De novo design of RNA and nucleoprotein complexes

Andrew Favor

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2026

Reading Committee:

David Baker, Chair

Frank DiMaio

Phil Bradley

Program Authorized to Offer Degree:

Molecular Engineering

University of Washington

Abstract

De novo design of RNA and nucleoprotein complexes

Andrew Favor

Chair of the Supervisory Committee:

David Baker

Department of Biochemistry

Nucleic acids fold into sequence-dependent three-dimensional structures and carry out diverse biological functions, much like proteins. However, while considerable advances have been made in the de novo design of protein structure and function, the same has not yet been achieved for RNA structures of similar intricacy. In this work, I describe the development of structure-generative diffusion models for generalized de novo biopolymer design, and I demonstrate their use in creating novel RNA folds and nucleoprotein complexes. With these tools in hand, I investigate design principles governing pseudoknot topologies and show how precise tertiary interactions can stabilize conformationally variable structures. I validate the robustness of this design approach through experimental characterization, demonstrating that designed sequences reliably self-assemble into their intended three-dimensional structures, and that engineered nucleoprotein complexes can be realized with high accuracy. Together, this work extends the principles of structure-based de novo protein design to nucleic acids and hybrid biopolymer assemblies, providing a foundation for creating a wide range of new structures and advancing the broader goals of molecular engineering.

Foreword

A lot has changed since I started this journey. The project described in this dissertation began relatively late in my PhD – a couple of months into my fourth year. Before then, I had worked on several projects tied together by a common theme: hierarchical design of multi-component nanomaterials. Those projects were fun, and I am proud of what I accomplished and learned during those years. However, over time, I became increasingly troubled by a sense of unease and dissatisfaction with my research: I did not feel intellectually challenged enough, and I did not feel like I was working on research that would adequately impact or inspire a wider scientific community. Eventually that feeling became strong enough that I knew I had to make a major change in project direction. Many people grow tired of their projects, but by year four of a PhD, the "sunk cost" starts to feel less like a fallacy and more like a very legitimate reality – pressure grows to finish the tasks in front of you because that seems like the most feasible path to the finish line, and starting over so late in the game could be detrimental.

Nonetheless, I chose to make a radical change, and switched to a project focused on developing machine-learning methods, which introduced me to entirely new sets of challenges and domains of knowledge. In this new direction, my specific focus (within a broader field of generative molecular AI) would deal with nucleic-acid design – a topic with which I had *zero* prior research experience.

Motivated by a deep desire to actualize my potential within the field of machine learning, and to express myself in a way that fully reflected my capabilities, I decided not to let this opportunity pass me by, and dove deep into that water.

While initially intimidating and perhaps risky, this became one of the best decisions of my life.

As it turns out, listening to one's inner calling can be profoundly important in changing course toward a brighter future. For me, that path led to the project discussed in this dissertation. It is not the only project of my PhD, but it is the one that I am most proud of. It has led to scientific opportunities I would otherwise not have experienced and has expanded my thinking in ways that feel deeply fulfilling. For narrative clarity, I do not include my earlier projects as dissertation chapters; instead, I focus this document on one central topic: extending the principles of *de novo* biomolecular design to nucleic acids and nucleoprotein complexes.

At the broadest level, the vision behind this work is to extend the structure-oriented molecular engineering that has been so successful in *de novo* protein design across the full central dogma of biochemistry. If we want to build molecular systems with capabilities approaching what we observe in nature, we must be able to *create* using the same full set of materials that natural systems use: proteins, RNA, DNA, and the higher-order assemblies they form together.

This dissertation describes initial steps toward that expansion. Although I most often reference the first implementation, *RFdiffusion-polymer*, the underlying generalization to biopolymer design has now been realized across three generations of the RFdiffusion framework. The approaches and results described herein therefore reflect fundamental engineering principles that I expect to remain evergreen as computational tools continue to evolve.

Contents

List of Figures

1

Introduction

1.1 Background and motivation

Noncoding RNAs adopt complex three-dimensional structures that underpin a wide range of biological functions, including genetic regulation, catalysis, and scaffolding of biochemical machinery. In the field of nucleic acid design, most strategies have focused on three main areas: aptamer generation via randomized sequence selection [1, 2] , the design of small RNAs with defined secondary structures[3, 4], and the construction of large, geometrically regular RNA/DNA origami [5, 6, 7, 8]. Aptamer generation has been largely structure-independent, while secondary structure-based approaches focus on two-dimensional base pair templates and do not account for three-dimensional geometry, which limits their utility in applications requiring precise spatial control. Three-dimensional structure design methods—such as nucleic acid origami, hierarchical motif alignments [9] , and parametric design based on propagation of idealized geometry [10, 11] —typically rely on rigid geometric assumptions, enforcing uniform helical axes and simplifying the placement of structural features like crossovers and base contacts. Previous methods tackle motif scaffolding, but rely on alignment of existing fragment libraries [12], rather than fully generating bespoke linkers, and are thereby more limited in the problems that they can be used to solve, especially if the fragment libraries are a subset of the full fold space of RNA. While these approaches can generate sophisticated and functional nucleic acid structures, they constrain the accessible design space and fail to capture the

complexity observed in native RNA folds, which often feature multi-angled junctions, bulges, bent helices, and loop-loop contacts, which are likely important for more complex functions. Thus, there remains a critical need for structure-oriented nucleic acid design approaches that enable exploration of the space of diverse three-dimensional structures exemplified by natural RNAs.

We reasoned that recent advances in generative deep learning for protein design could be leveraged to address the current limitations in nucleic acid design. Generative models such as RFdiffusion have shown considerable success in structure-oriented protein design [13, 14, 15] , and the RoseTTAfold framework has been adapted for nucleic acid structure prediction [16, 17]. Although denoising diffusion probabilistic models (DDPMs) have been applied to generate short RNA monomers [18, 19, 20, 21, 22] , to date, these efforts lack hierarchical design capabilities, lack multi-polymer structure generation, and none have yet been experimentally validated. We set out to generalize the RFdiffusion de novo protein design approach to nucleic acids, and to explore the use of this method by designing novel RNA structures and protein-nucleic acid assemblies.

1.2 Computational approach

Our design approach proceeds in two steps. First, we generate RNA or DNA backbone structures using an extended version of RFdiffusion, RFdiffusion-polymer (RFDpoly), described in the following paragraphs. Second, we design base sequences on the generated backbones using NA-MPNN [23] , which generalizes the widely used ProteinMPNN [24, 25] to nucleic acids.

RFDpoly extends RFdiffusion to enable the generation of nucleic acid structures by building on RF2-AllAtom [16] [17] , which models both proteins and nucleic acids and accepts conditional information through 1D, 2D, and 3D channels. Previous RFdiffusion implementations could only denoise protein structures around fixed nucleic acid ligands; RFDpoly generates nucleic acid structures as well. Unlike proteins, where all non-frame backbone atoms (O, Cβ) can be deterministically placed given frame coordinates, nucleic acid backbones have increased conformational freedom due to variable sugar and phosphate geometries. To address this, RFDpoly still denoises frame rotations

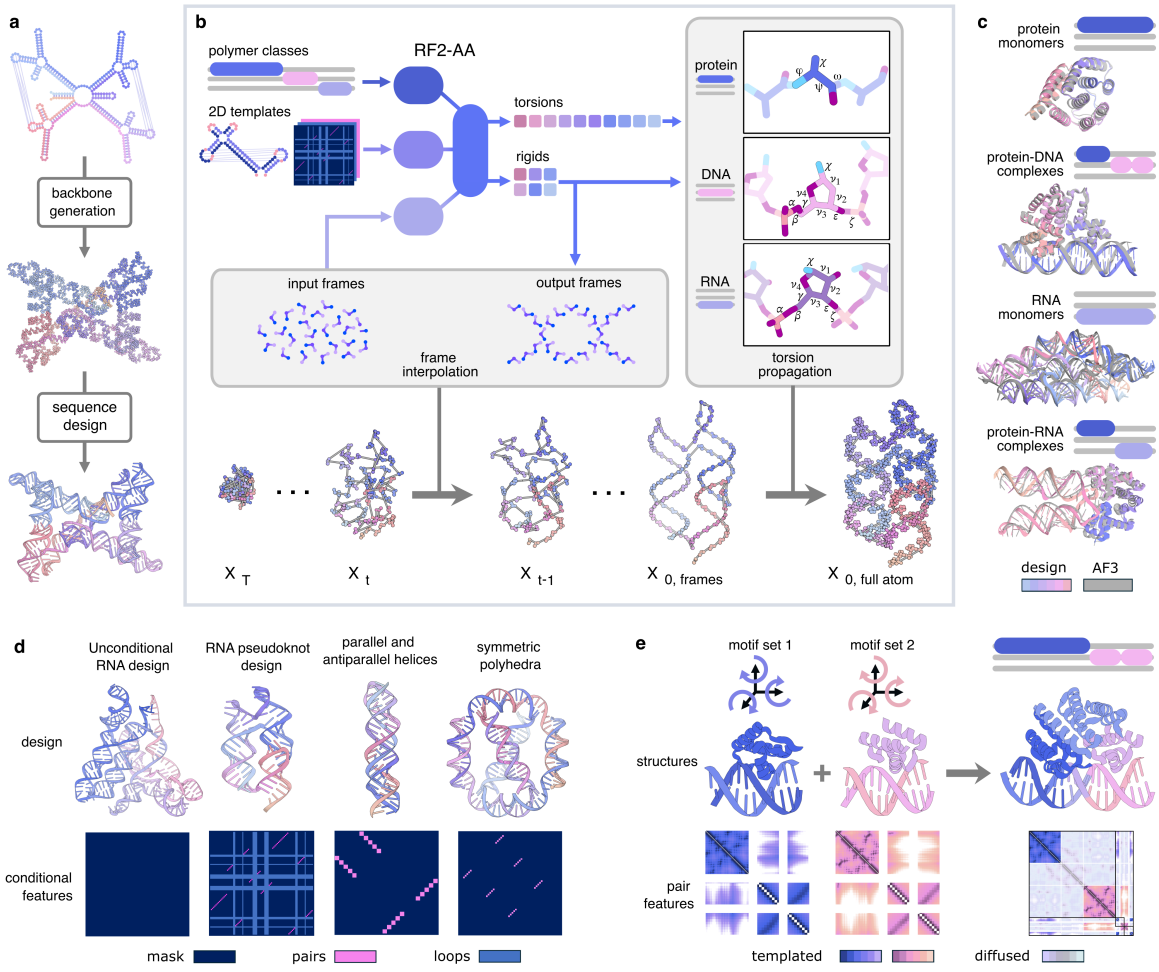

Figure 1 | Generalized biopolymer structure generation a, Standard de novo structure design pipeline, consisting of backbone generation followed by sequence design. **b**, RFDpoly denoises backbone frames using RoseTTAfold's 3D track, while conditional information from the 1D and 2D tracks (such as polymer class and base pair patterns, respectively) guides the generation toward desired structures. Full sugar–phosphate backbone coordinates are constructed by propagating torsion angles about frames, using connectivity graphs defined by sequence predictions. **c**, Polymer-class labels provided at inference instruct the model to generate chemically accurate structures appropriate for each molecule type. In silico predictions with AlphaFold3 show high self-consistency for proteins, DNA, and RNA. **d**, Structural control via base pair templating enables RFDpoly to generate nucleic acid structures with diverse topologies. User-specified controls define templates for RoseTTAfold's 2D track. **e**, Hierarchical design in RFDpoly uses 2D templates for motif structure, allowing global placement to be inferred during the denoising process.

and translations as in previous RFdiffusion models, but goes further by using rigid body parameters and torsion angles predicted from RoseTTAfold to construct complete nucleic acid backbones (and side chains, when sequence or motif geometry is provided). To increase the extent to which the denoised frames determine nucleobase positioning, we switched from the phosphate-centered frame atoms used in RF2-AllAtom (OP1-P-OP2) to O4'-C1'-C2' frame atoms, which are closer to bases.

To condition the diffusion process, RFDpoly uses one-hot encoded molecule-class features, provided through RoseTTAfold's 1D track, to guide the generation of appropriate backbone geometry for each polymer type – producing chemically accurate RNA and DNA without compromising protein structure quality. With RFDpoly trained to denoise these different biopolymers, we can specify which polymer class is generated in each contiguous chain during inference (Fig. 1c).

To enable RFDpoly to condition on base pairing networks when generating 3D RNA structures, we expanded RoseTTAfold to accept base pair partner labels through its 2D-track (Fig. 1b). These conditional features allow RFDpoly to generate backbones that possess specified secondary structure patterns. To enable user guidance over structure generation in ways that are relevant to nucleic acid design, we provide several routes for controlling secondary structure features at inference time (Fig. 1d). Secondary structure strings (in the form of dot bracket notation) can be provided to generate structures with specific pseudoknot topologies, allowing users to generate 3D models for RNA puzzles [3, 26, 27] or other sources of secondary structure templates. Beyond generating canonical single-partner base pairs, users can specify whether pairs are parallel or antiparallel, giving precise control over tertiary structure features such as triple helices or G-quadruplexes. Lists of paired sequence regions can be specified, with cyclic- or repeat-symmetry, to reflect design patterns often seen in nucleic acid origami.

We train RFDpoly using a randomized masking scheme in which training examples are divided into motif regions (containing ground truth structure information) and diffused regions – to emulate design tasks such as motif-bridging, docking, or generating new structure around central motifs. To enable RFDpoly to generate both protein and nucleic acid structures, and assemblies containing both polymer types, we train the model using datasets consisting of proteins, RNA, and DNA

either as monomers or as mixed polymer complexes. While previous versions of RFdiffusion only generate structure within protein chains, keeping nucleic acids as fixed motifs, the mask generators for RFDpoly treat all polymer classes equally, denoising noised regions within nucleic acid chains as well as proteins.

Once backbone structures have been generated, sequence design is performed using two methods. In the first approach, NA-MPNN is used to assign sequences to complete diffusion-generated backbones [23] , and PyRosetta is then used to build and repack sidechains. In the second approach, sequence-structure codesign, RoseTTAfold sequence predictions are autoregressively decoded along with rigid-body and torsion parameters to produce full atom sidechain representations. Sequences designed with NA-MPNN were predicted to fold with better self-consistency metrics, but the presence of side chains from codesign trajectories allowed immediate filtering of structures for target base pair pattern satisfaction, without going through the process of MPNN-design and PyRosetta repacking.

Designed sequences were filtered based on similarity between design models and predicted structures at multiple levels. Initial filtering was based on predicted secondary structure self-consistency using RibonanzaNet[28] ; similarity between designed and predicted secondary structure was evaluated using F1-scores[29] . Tertiary structures of designs with high F1 scores were predicted using Chai-1[30] or AlphaFold3[31], and compared to design models, using lDDT[29, 32, 33] or TM-score [34] as metrics for structure similarity. We use the term "designable" below to describe designs for which the predicted structure matched the design model.

In unconditional structure generation calculations, the generated structures maintained high designability across increasing lengths, with little drop-off in secondary structure self-consistency metrics, and minimal steric clashing even in large-scale generation tasks (Extended Data Fig. 2a). Tertiary structure diversity was evaluated by comparing inter- and intra-group distributions of 3-dimensional fold similarities for diffusion-generated structures against native RNAs from structural databases, or against other diffusion-generated structures; tertiary structure designability was evaluated by comparing designed RNAs to AF3 predictions of their MPNN-designed sequences. Pair-

wise alignment of RNA structures (within 20% sequence length) from both generated and native sets (˜1,700 RCSB RNA structures) using USalign[34], yielded distributions of tertiary structure similarity (given by TM-score) across various length scales (Extended Data Fig. 2b). While the generated structures of short RNAs had folds resembling those in the PDB (e.g., tRNAs), diversity increased rapidly for structures longer than 120 bases. RFDpoly-generated structures were dissimilar from native RNAs, while still having high AF3-predicted self-consistency (Extended Data Fig. 2c), indicating that the model can explore novel folds on the global level, without loss in designability.

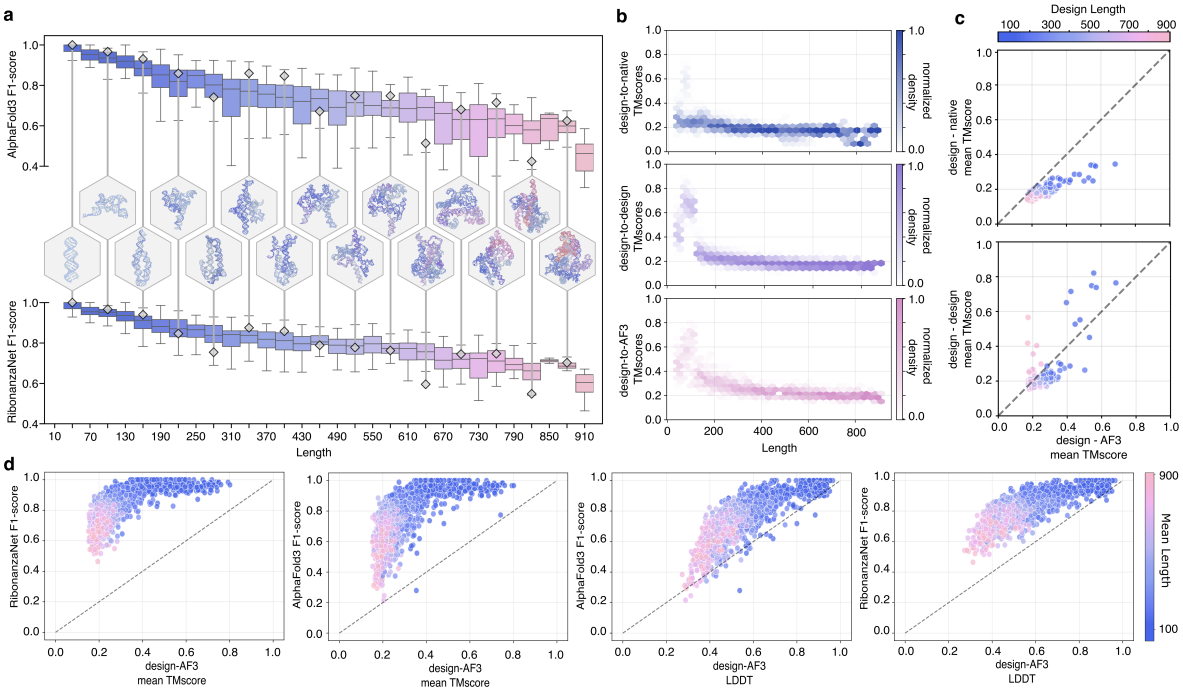

Figure 2 | In silico performance of unconditional RNA structure generation a, Predicted secondary structure accuracy as a function of sequence length. Top: F1-score similarity between AlphaFold3 predictions and design model secondary structures. Bottom: F1-score similarity between RibonanzaNet predictions and design model secondary structures. Representative designs are shown in hexagons, with vertical connectors to their corresponding F1-score markers. **b**, Distributions of pairwise TM-scores across different length ranges (coverage threshold 0.8). Top: RFDpoly-generated design models aligned to native RNAs. Middle: designed structures aligned to other designed RNAs. Bottom: designed structures aligned to AF3-predicted RNAs. **c**, Per-design comparison of TM-scores to show structural similarity between groups vs each design's AF3-aligned TM-scores as a self-consistency metric (shared x-axes). Top y-axis: TM-scores of designed backbones aligned to similar-length native structures. Bottom y-axis: TM-scores of designed backbones aligned to other designed backbones. **d**, Comparison of in silico predicted secondary structure accuracy metrics (y-axes, RibonanzaNet F1-score or AF3 F1-score) with tertiary structure accuracy (x-axes, TM-score or LDDT from alignment to AF3 model).

2

Experimental evaluation in the Eterna OpenKnot challenges

To benchmark the ability of our method to generate structures with specific base pair patterns, we participated in the Eterna OpenKnot challenges [27]. Base pairing templates from a set of 57 pseudoknot puzzles were used to condition denoising trajectories (Fig. 3a), and sequences were assigned using NA-MPNN. For all puzzles attempted, reference wild-type sequences were provided for comparison, and many puzzles had experimentally determined 3D structures, which were also redesigned with NA-MPNN. To experimentally characterize the secondary structures of designed sequences, the OpenKnot organizers used SHAPE-seq, which generates reactivity profiles that distinguish between base pairing and non-base pairing regions [28, 35, 36, 37].

2.1 Template-conditioned pseudoknot generation

For clarity, we first describe the design approach and the comparison to experimental data in more depth for a specific case, puzzle Q-19 (SV_r7_240_4), which consists of five hairpins ($\alpha 1$–$\alpha 5$). In the target structure, hairpin pairs ($\alpha 1 : \alpha 3$) and ($\alpha 4 : \alpha 5$) form kissing-loop interactions, while hairpin $\alpha 2$ remains unpaired (Fig. 3a, left). Starting from random noise, the base pair-conditioned diffusion denoising trajectory progressively builds an RNA structure adhering closely to these predefined

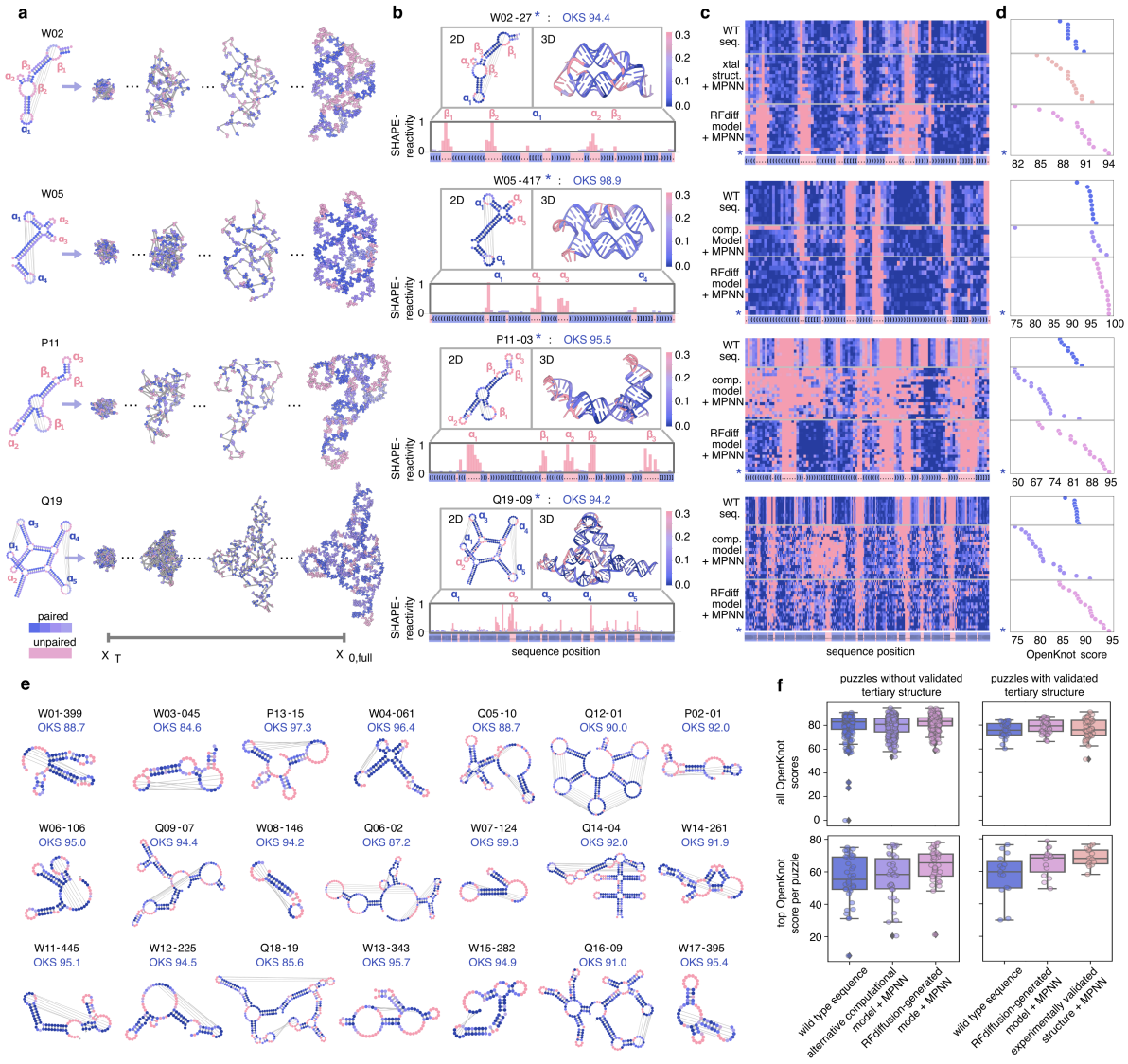

Figure 3 | Experimental validation with SHAPE-seq a, Denoising trajectories for representative RNA puzzles, where puzzle templates specify base pair patterns and unpaired loop regions. **b**, SHAPE profiles for selected designs show high reactivity in unpaired regions (structures colored by reactivity), consistent with target secondary structure templates. Annotated structural features include hairpins (α) and bulges (β). **c**, Experimental SHAPE profiles compiled for wild-type sequences, MPNN-redesigns of alternative backbones (experimental or computational), and MPNN-designs of diffusion outputs. Selected designs (*) best matched target structures. **d**, Target OpenKnot scores derived from SHAPE profiles. **e**, SHAPE profiles mapped onto puzzle secondary structures with corresponding OpenKnot scores, for selected puzzles. **f**, Distribution of all Target OpenKnot scores (top) and best scores achieved per puzzle (bottom), separated by puzzles with experimentally determined 3D structures (left) and those with only computational models (right).

contacts and hairpin geometries (Fig. 3a, right).

2.2 Experimental assessment: SHAPE-seq

Hosts of the OpenKnot competition experimentally assessed this design using selective 2'-hydroxyl acylation analyzed by primer extension sequencing (SHAPE-seq), which reports on nucleotide flexibility and accessibility by chemically modifying unpaired or flexible nucleotides. Modified nucleotides impede reverse transcription, allowing sequencing readouts to quantify site-specific reactivities. Regions displaying high SHAPE reactivity correspond to unpaired or loop nucleotides, whereas paired bases are protected from modification and thus exhibit low reactivity. To show how reactivity data map back to structural features, we color secondary and tertiary structures by reactivity profile for a selected design (*) (Fig. 3b). Consistent with expectation, we observe low reactivity in the paired kissing loops, but high reactivity in the unpaired stem loop bases ($\alpha 2$). Thus, the SHAPE-seq reactivity profiles indicate that the design secondary structure closely matches the target secondary structure (horizontal axes, with desired paired and unpaired positions colored in blue and red, respectively). To systematically evaluate performance across multiple design strategies, SHAPE-reactivity profiles were compiled into two-dimensional heatmaps (Fig. 3c), comparing our diffusion-based method with alternative backbone redesigns and wild-type RNA sequences. The selected design (*) displayed optimal agreement with the target structure, as quantitatively captured by the highest achieved target OpenKnot score among designs for puzzle Q-19 (Fig. 3d).

Overall, SHAPE-seq experimental reactivity profiles for targeted puzzles were close to the input base pairing profiles (Fig. 3c). Target OpenKnot Scores [38] were used as an aggregate metric to summarize how well each design folded into the target secondary structure (Fig. 3d-e). For most puzzles attempted, at least one sequence from MPNN-design of RFDpoly-backbones outperformed the wild-type sequences and MPNN-redesign of experimental reference structures (Fig. 3f, Extended Data Fig. 5a-c).

2.3 The span of structural control through base pair templating

These results show that RFDpoly can generate 3D models across a wide range of target secondary structures. User-input enables the generation of backbones for a diverse set of pseudoknot topologies without loss in designability (Fig. 3e). Many of the designs characterized by SHAPE-seq display highly accurate formation of intricate non-standard tertiary elements. For example, P15-07 and P20-10 exhibit perpendicular helix–helix groove docking, Q17-07 and Q16-09 form high-order multi-way junction hubs, and W08-146 and P05-14 use loop–groove clasping interactions to stabilize packing (Extended Data Fig. 4). The diversity and complexity of secondary structures designed indicate that RFDpoly can explore a complex fold space beyond that which is accessible to origami approaches or even observed in native systems.

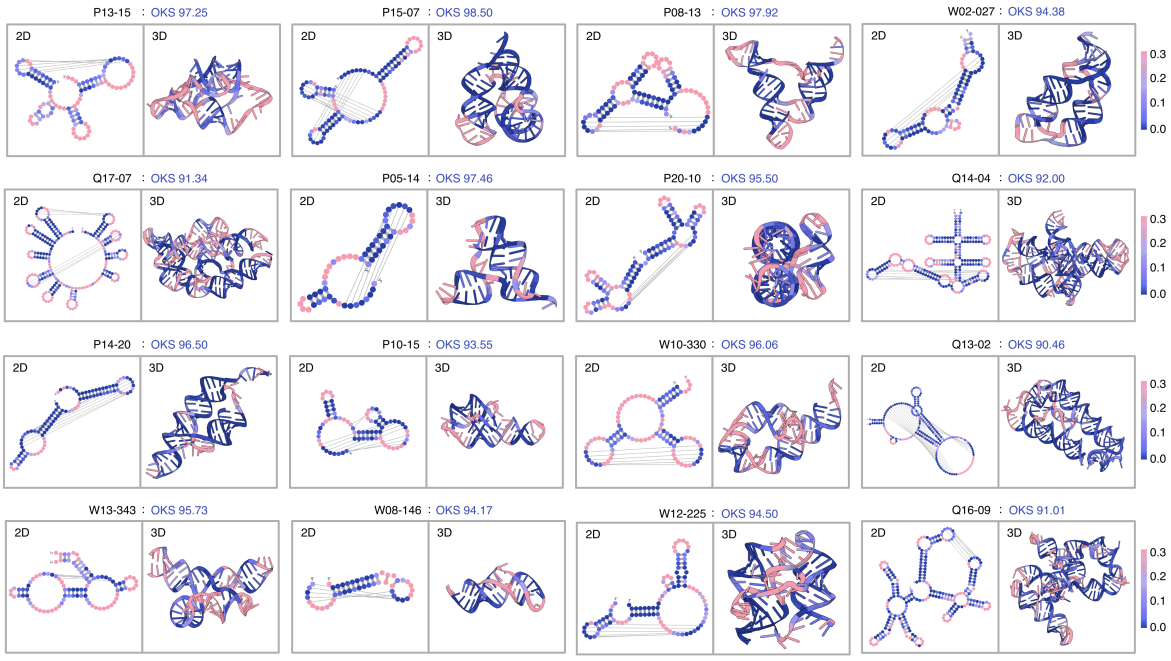

Figure 4 | Topologically diverse RNA folds designed with high experimental agreement Diverse and intricate RNA folds from OpenKnot competition designs, represented as secondary and tertiary structures, designed with high accuracy according to SHAPE-seq. Examples include: kissing loops, hairpin-strand contacts, precise bulge placement, triple-strand contacts, multi-junctions.

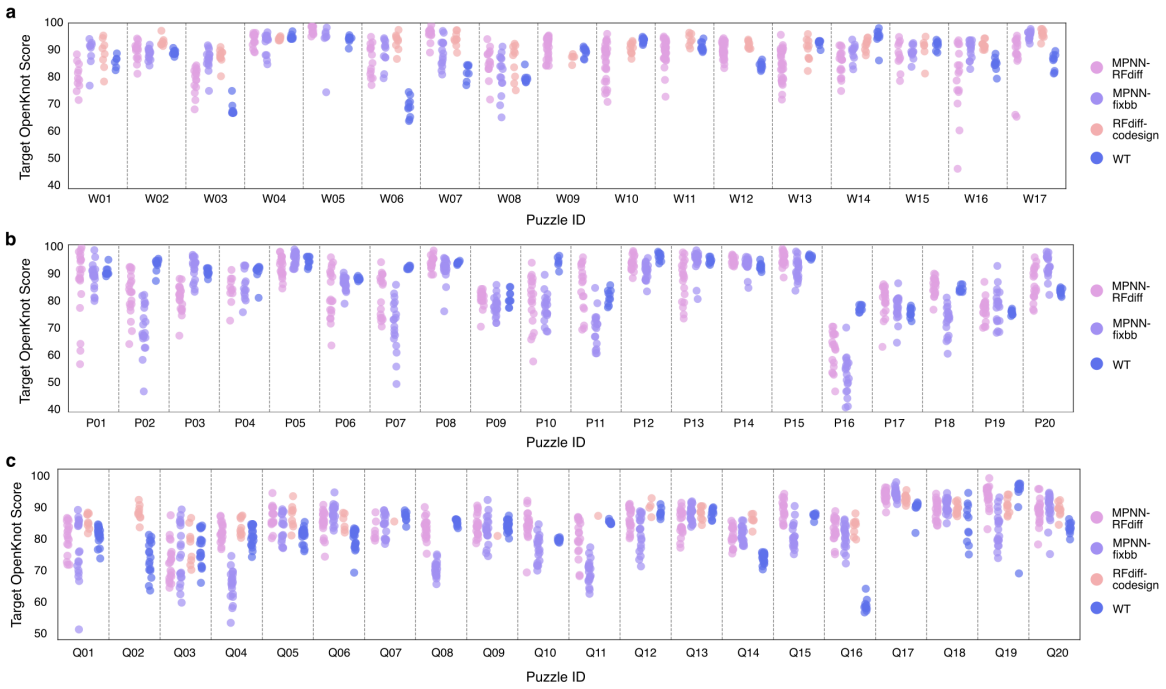

Figure 5 | Target OpenKnot scores across competition rounds a, OpenKnot round 6. **b**, OpenKnot round 7a. **c**, OpenKnot round 7b. Note: MPNN-fixbb refers to both experimental structures and alternative (non-RFDpoly generated) computational models redesigned by MPNN, depending on the puzzle and what alternative structures were available for each puzzle. RFdiff-codesign refers to the autoregressive decoding of RFDpoly's own sequence predictions over the last 40 steps of 50-step denoising trajectories.Extended Data Figure 3: Nonstandard RNA folds.

3

Design of pseudocycles

To further assess RFDpoly's design capabilities, we sought to create larger novel RNA architectures. We focused on designing RNA pseudocycles: large, monomeric RNA structures composed of nearly cyclically repeating subunits connected in a single continuous chain.

3.1 Large pseudocycles

These folds were chosen as design targets because their large size, relative to our previously designed RNA pseudoknots, would enable characterization via electron microscopy, and their nearly symmetric morphologies would be easily identifiable during screening. We chose single-chain pseudocycles rather than cyclic assemblies to avoid the complications of multimeric assembly, enabling direct application of in silico screening metrics optimized for single-chain RNAs.

To generate large pseudocycles with numerous intra-chain contacts to reduce structure flexibility, we developed a secondary structure template generation protocol that accepts graph-based definitions of regional pairing and efficiently searches for fitting base pair patterns, diversified using random sampling of subregion lengths (Fig. 6a). Once candidate base pair templates were generated, they were provided to RFDpoly as 2D templates to guide the denoising process. Pseudocyclic RNA backbones were generated using symmetric noise propagation during denoising, while maintaining backbone connectivity between subunits in encoded bond features. After backbone

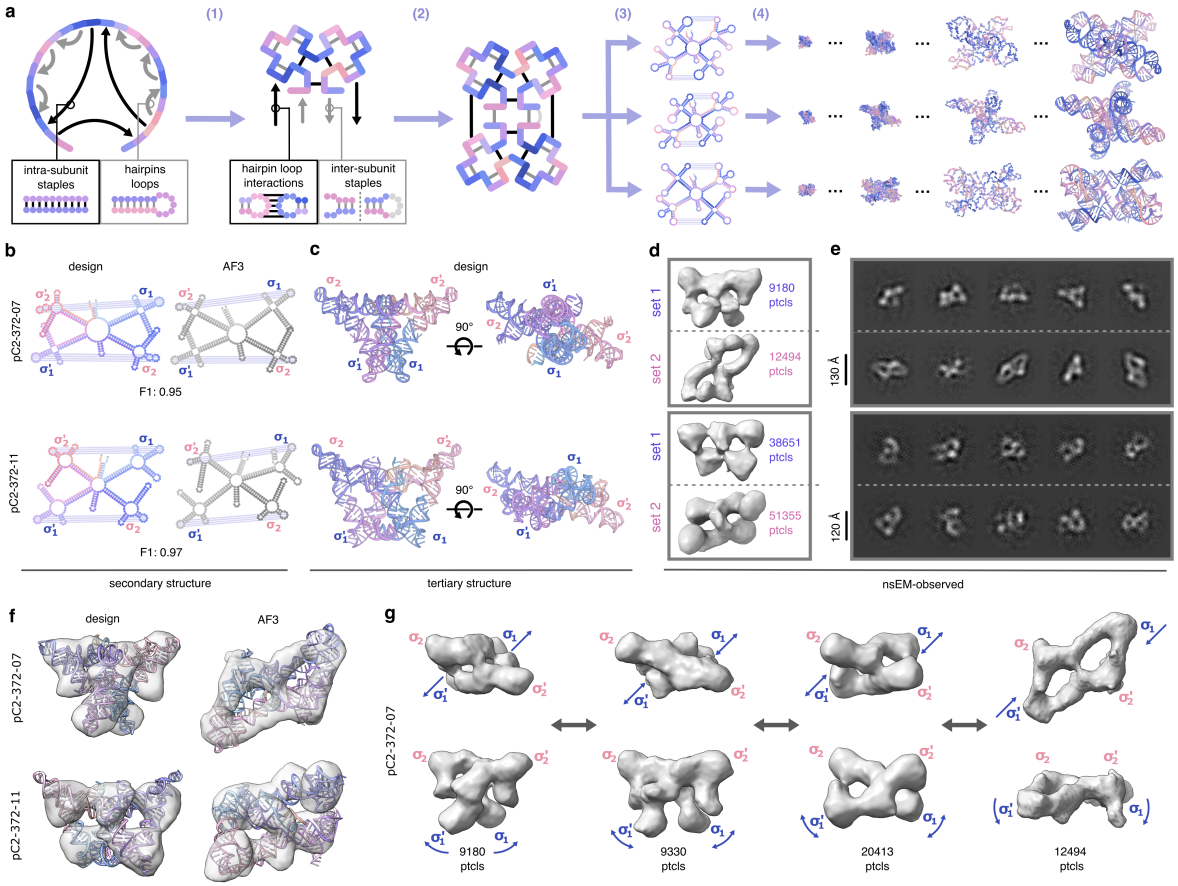

Figure 6 | Design and characterization of 372-base pseudocycles a, Workflow for generating base pair templates and RNA pseudocycles: (1) initialize subunit segments and assign intra-subunit partner regions; (2) propagate subunits symmetrically and assign inter-subunit pairs; (3) randomly vary segment lengths to diversify structures; (4) run RFDpoly denoising with secondary structure conditioning. **b**, Comparison of secondary structures and F1-similarity scores between design models and AlphaFold3 predictions for selected designs (top: pC2-372-07, bottom: pC2-372-11). **c**, Tertiary structures of selected designs. **d**, 3D reconstructed nsEM volumes showing butterfly-like folds consistent with design models and AF3 models. **e**, 2D class averages from nsEM data. **f**, Design models and AF3 models fit into reconstructed volumes. **g**, Conformational variability in pC2-372-07 visualized across multiple nsEM volumes.

generation and sequence design, in silico predictions of tertiary structure with AlphaFold3 showed conformational variability between models, while having matching secondary structures between designed and predicted structures (Fig. 6b-c). Synthetic genes were obtained and in vitro-transcribed for eight 372-base designs, all sharing a common twofold pseudocyclic symmetry separated by hinge-like loops.

Characterization with negative stain electron microscopy (nsEM) revealed well-formed C2 pseudosymmetric structures for six designs (Extended Data Fig. 4a-c). Observed morphologies were polydisperse, with the butterfly-like folds sampling multiple conformational states, and the two rigid side-domains folding about their central hinge loops. For two of the four designs (pC2-372-07, pC2-372-11), class averages for one of the primary observed particles (denoted set-1) matched the design model, while others (set-2) more closely matched the AlphaFold3 predicted models (Fig. 6d-f). For the remaining designs, reconstructed volumes did not closely resemble either the design or predicted structures, while nsEM revealed alternative pseudosymmetric folds (Extended Data Fig. 4d). All of the nsEM-characterized designs had similar secondary structure features (defined by four characteristic cloverleaf folds: σ1, σ2, σ1', σ2'), and for each design with nsEM 3D reconstructions, one of the primary observed states had the target double-triangle fold (set-1). Fitting of design models into 3D-reconstructed volumes for particle sets-1 demonstrated structural consistency within this subpopulation (Fig. 6f). To further characterize the hinge-motion between the open and closed conformations, we classified particles into multiple 3D volumes, which revealed connectivities between the designed conformation and alternative states (Fig. 6g).

3.2 Compact pseudocycles

We next designed smaller (240 nt) compact RNA pseudocycles, aiming for well-defined tertiary structures by enriching for supportive base pair contacts and iteratively refining the overall structure by recycling the most well-ordered regions of the models back into RFDpoly as base pair templates, over multiple rounds of design and selection. These shorter-length sequences could

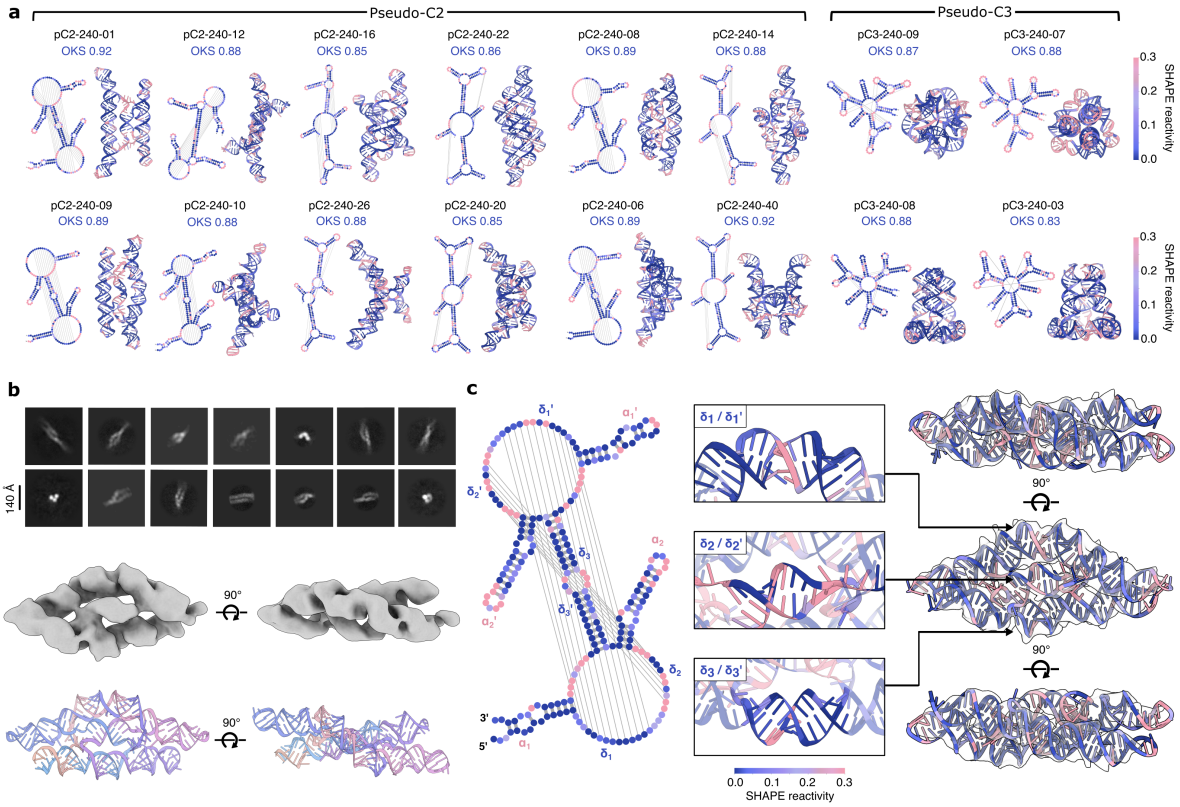

Figure 7 | Characterization of small, compact pseudocycles a, Representative compact pseudocycle designs; secondary (left) and tertiary (right) structures colored by SHAPE-seq reactivity. **b**, Cryo-EM 2D classes (top), and 3D reconstructed volume (middle) of designed pseudocycle, pC2-240-04, revealing the overall fold of the design model (bottom). **c**, SHAPE reactivity profile of pC2-240-04, mapped onto the secondary structure (left) and tertiary structure (right), refined to fit into the cryo-EM volume. Three central helical domains ($\delta 1$, $\delta 2$, $\delta 3$) are highlighted (middle), showing crossover base pairs in the secondary structure and corresponding strand interweaving in the tertiary fold.

be characterized with SHAPE-seq, allowing us to compare experimental results between chemical probing and electron microscopy. Fifty designs with two- or three-fold pseudocyclic symmetry were selected for SHAPE-seq characterization based on high in silico self-consistency metrics. Overall, there was strong agreement between experimental reactivity profiles and target secondary structures (Fig. 7a), with 42 designs achieving Target OpenKnot Scores above a target threshold of 0.8 (Extended Data Fig. 8a). Of these, 19 were selected for further characterization by electron microscopy, based on whether Chai-1 [30] or AlphaFold3 [31] predictions resembled the design models. RNA transcribed from DNA templates was screened by nsEM, and the most structured candidates were further characterized by cryo-EM. Three designs (pC2-240-02, pC2-240-03, pC2-240-04) had good SHAPE-reactivity profiles (Extended Data Fig. 8b) and well-defined 2D classes from cryo-EM (Fig. 7b, top; Extended Data Fig. 8c-d).

For pC2-240-04, cryo-EM characterization led to a 6.6 Åreconstructed volume, with clearly observable helical pitch and a global fold matching the design model (Fig. 7b, middle and bottom). The experimental SHAPE-reactivity profile matches the designed secondary structure (Fig. 7c, left), and the design model could be fit into the cryo-EM reconstructed volume (Fig. 7c, right). The locations of paired and loop regions observed in the cryo-EM reconstruction are consistent with the SHAPE-reactivity profile. Of particular interest are three central helical domains, $\delta 1$, $\delta 2$ and $\delta 3$ (Fig. 7c, middle), with crossover regions in the secondary structure that make this fold a complex high-order pseudoknot (since these central crossover domains can occupy separate locations in 2D secondary structure diagrams, we label paired strands as δi and δi‘, denoting the upstream and downstream components, respectively). The interweaving of these three helical domains between protruding helical hairpins in the design model is confirmed in the 3D tertiary structure. To assess the novelty of this fold, we aligned the experimentally refined model against all representative RNAs from structural databases (see Methods) and found no close matches: the best hit was to a small region of the plant mitochondrial ribosome (PDB 6XYW, chain A) with a TM-score of 0.329 at 0.55 coverage. The combined validation using SHAPE-seq and cryo-EM shows that RFDpoly can accurately create intricate new RNA tertiary structures.

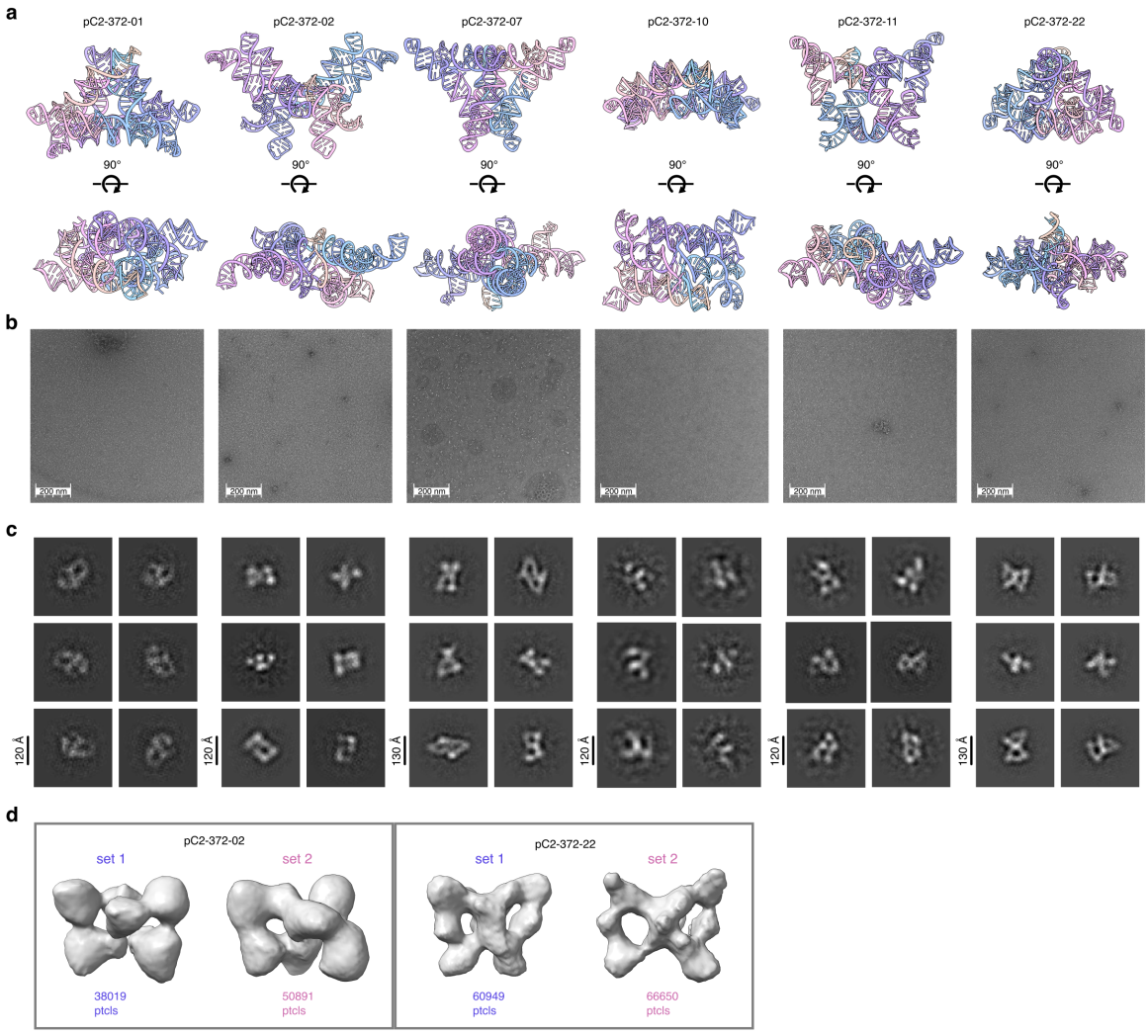

Figure 8 | nsEM data for butterfly fold (372 nt) pseudocycles Data for six designs (pC2-372-01, pC2-372-02, pC2-372-07, pC2-372-10, pC2-372-11, pC2-372-22). **a**, Design models shown in side-views and top-down views. **b**, Representative negative stain micrographs. **c**, 2D class averages. **d**, Example 3D reconstructed volumes for classes corresponding to two conformationally-distinct particle sets, shown for pC2-372-02 (left) and pC2-372-22 (right).

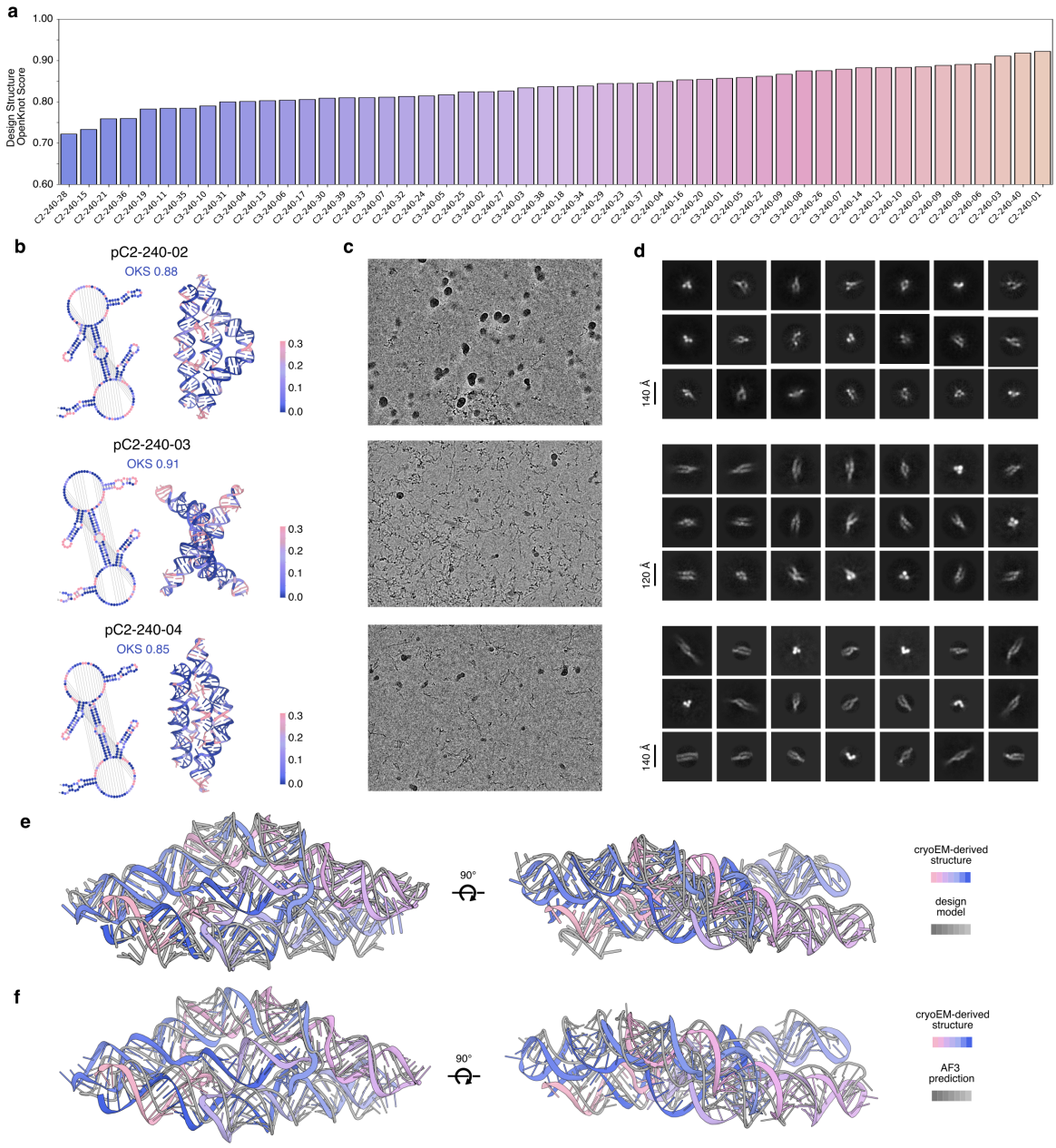

Figure 9 | Experimental data for compact (240 nt) pseudocycles a, Target OpenKnot scores (agreement of SHAPE reactivity with design-model secondary structures) for all 50 compact pseudocycle designs. **b**, Selected designs (pC2-240-02, pC2-240-03, pC2-240-04); secondary and tertiary structures colored by SHAPE-seq reactivity. **c**, Representative cryo-EM micrographs, and **d**, 2D class averages for selected designs. **e**, Comparison of the design model and ISOLDE-refined model [39] , aligned by USalign (RMSD=5.35 Å{}, TM-score=0.58). **f**, Comparison of the best fitting AlphaFold3 model and ISOLDE-refined model (see Methods), aligned by USalign (RMSD=5.64 Å{}, TM-score=0.51).

4

Motif scaffolding and hierarchical design

We next set out to use RFDpoly to design protein–nucleic acid assemblies. We began by incorporating cognate protein and DNA motif fragments from previous work [40] into larger assemblies; simultaneously fusing interacting protein (C-to-N) and DNA (3'-to-5') motifs with newly generated rigid connective structures. We used 2D motif-template conditioning to embed three sequence-orthogonal DNA-binding proteins (DBPs) and their respective target DNAs (Fig. 10a) into larger connected assemblies, allowing their global placement and the connections between them to be inferred during the denoising process (Fig. 10b). Base pair templates that specified DNA strand exchange were used to condition the denoising process, and control long-range connectivity (Fig. 10c,d), enabling the creation of intricate nucleoprotein complexes with cooperative, multi-contact binding surfaces that would be challenging to achieve with purely parametric approaches.

4.1 Strand-exchanging complexes

We first applied this hierarchical strategy to design protein–DNA complexes in which protein homodimers bind to strand-exchanging DNA heterodimers to form hybrid complexes reminiscent of classical Holliday junctions [41] (Fig. 10e). These designs were created using inter-helical fusions (Fig. 10b, top) which connect two DBPs across different bound helices, acting as rigid "staples" that lock the relative orientation of the DNA helices, reflecting prior strategies in protein-DNA

hybrid nanostructures [42] . An initial set of designs was generated in which DBP fusions form bridges across DNA helical extensions; upon characterization by nsEM, these designs showed high conformational variability, likely due to limited buttressing interactions, which complicated structural characterization. Despite uncertainty in the geometries of these complexes, we did observe on-target trimer assembly for one design, ECC09 (Fig. 10e-g, top row). Difficulties in characterizing the flexible complexes motivated a second design campaign, in which the generated protein structure not only connects DBPs but also forms a homooligomeric interface with the other protein chains in the complex, to create a more rigid protein-DNA assembly. This second round of protein-protein interface-forming complexes was expressed and again characterized by nsEM, revealing successful assembly for two designs, ECC36 and ECC38 (Fig. 10e-g, bottom two rows). The pseudo-symmetric structure of these complexes is shown clearly in 2D class averages (Fig. 10f) and 3D reconstructed volumes (Fig. 10g), consistent with the bent configuration of the DNA in these strand-exchanging complexes. Thus, RFDpoly can carry out motif scaffolding while simultaneously generating designs with complex secondary structures, providing a route to bridge the worlds of protein design and DNA origami.

4.2 Linear fusions

We next used the same flexible scaffolding approach to generate new proteins predicted to bind extended contiguous DNA sequences (Fig. 10h). This second hierarchical design campaign utilizes intra-helix DNA fusions (Fig. 10b, bottom) in which DBPs are concatenated along a shared helical axis to create a longer sequence-specific recognition domain, expanding on existing techniques for the modular assembly of DNA-binding proteins[43, 44]. By sampling the geometry of the connecting DNA during DBP fusion, the designs can bind DNA in a wide range of conformations, in contrast to a previous approach limited to linear, parametrically generated B-form DNA [40] . The newly designed intra-helix DBP fusions were evaluated by yeast-display flow cytometry experiments (Fig. 10h, bottom) in which biotinylated dsDNA targets were titrated to generate

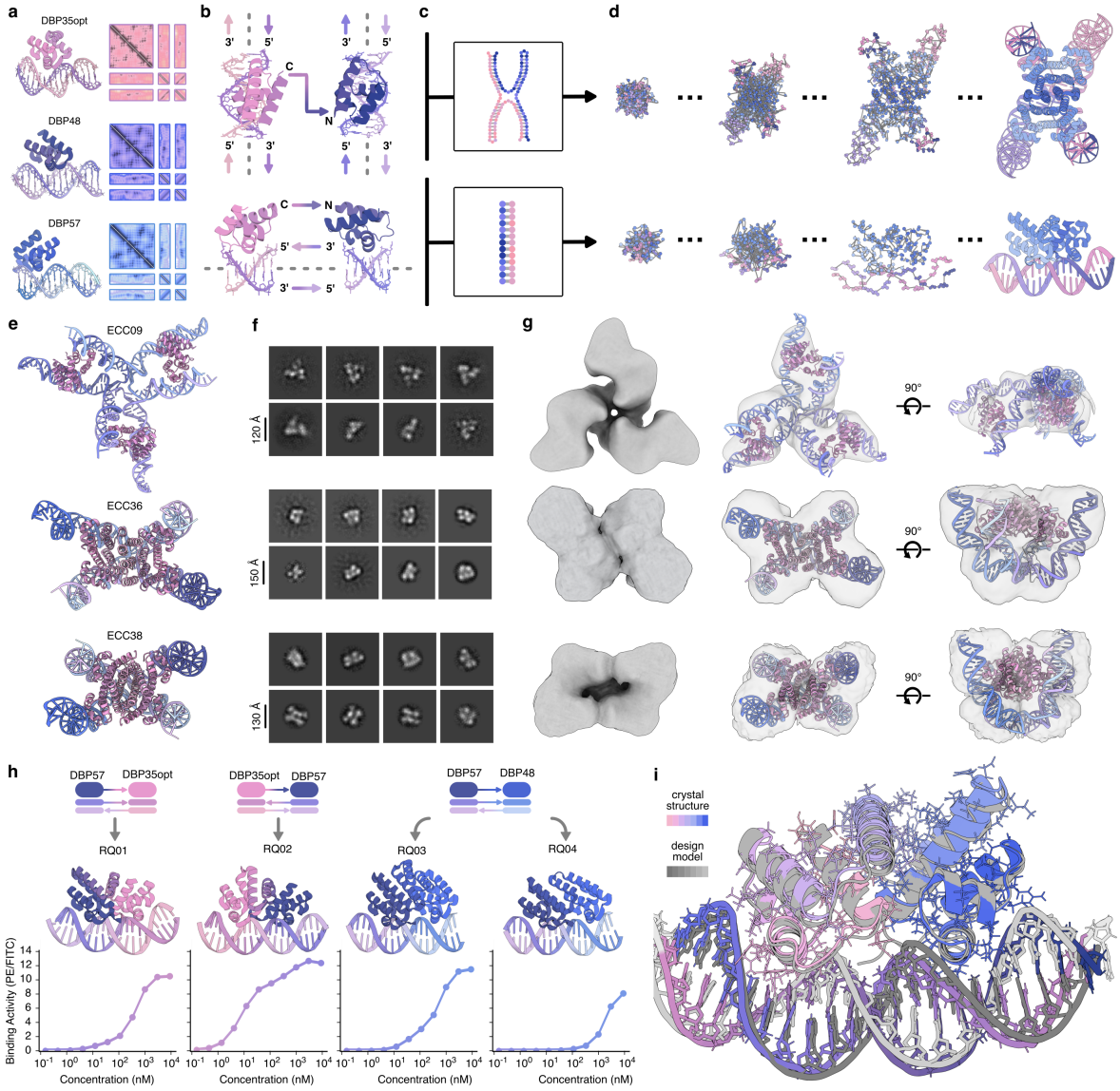

Figure 10 | Multi-polymer fusions and motif scaffolding **a**, DNA-binding proteins (DBPs) and their 2D templates, provided to RFDpoly for motif scaffolding. **b**, Multi-chain protein-DNA fusions: either inter-helical fusions bridging two DBPs across different helices (top) or intra-helical fusions parallel to the DNA axis (bottom). **c**, base pair pattern specification enables topological control over generated DNA interfaces. **d**, Denoising trajectories run with or without symmetry (top and bottom, respectively) to generate complex assemblies. **e**, Design models of three nucleoprotein junctions, with protein domains bound to strand-exchanging DNA hetero-oligomers. **f**, nsEM 2D class averages of selected designs. **g**, nsEM 3D reconstructed volumes shown alone (left), or superimposed over design models (middle and right); final maps refined using symmetry constraints. **h**, Intra-helical fusion of different DBPs in various binding orientations (top) generated larger sequence-specific DNA-binding proteins (middle), which bind to longer stretches of target DNA, as evaluated by yeast-display flow cytometry assays (bottom). Binding quantified as phycoerythrin/FITC ratios. **i**, Crystal structure of DBP-fusion RQ-01 superimposed on the design model using USalign[34] (RMSD = 1.76 Å{}, TM-score = 0.90).

binding curves, yielding apparent KD values from ˜2 μM (RQ-04) to ˜10 nM (RQ-02).

We assessed model accuracy by X-ray crystallography, obtaining a 3.0 Å{} resolution crystal structure of RQ-01 bound to its DNA target (Fig. 10i). The crystal structure closely matches the design model over the full assembly, including the rigid linkers; notably, the bent DNA conforms to the design, indicating that RFDpoly can accurately design non-ideal nucleic-acid geometries. Together, these results showcase the utility of RFDpoly for generating new DNA-binding proteins specific for extended DNA sequences and the ability to scaffold multiple biopolymer types simultaneously. Overall, our hierarchical design approach allows for the reuse of previously characterized structures to accelerate further design efforts and enables the construction of increasingly complex classes of biomolecular architectures.

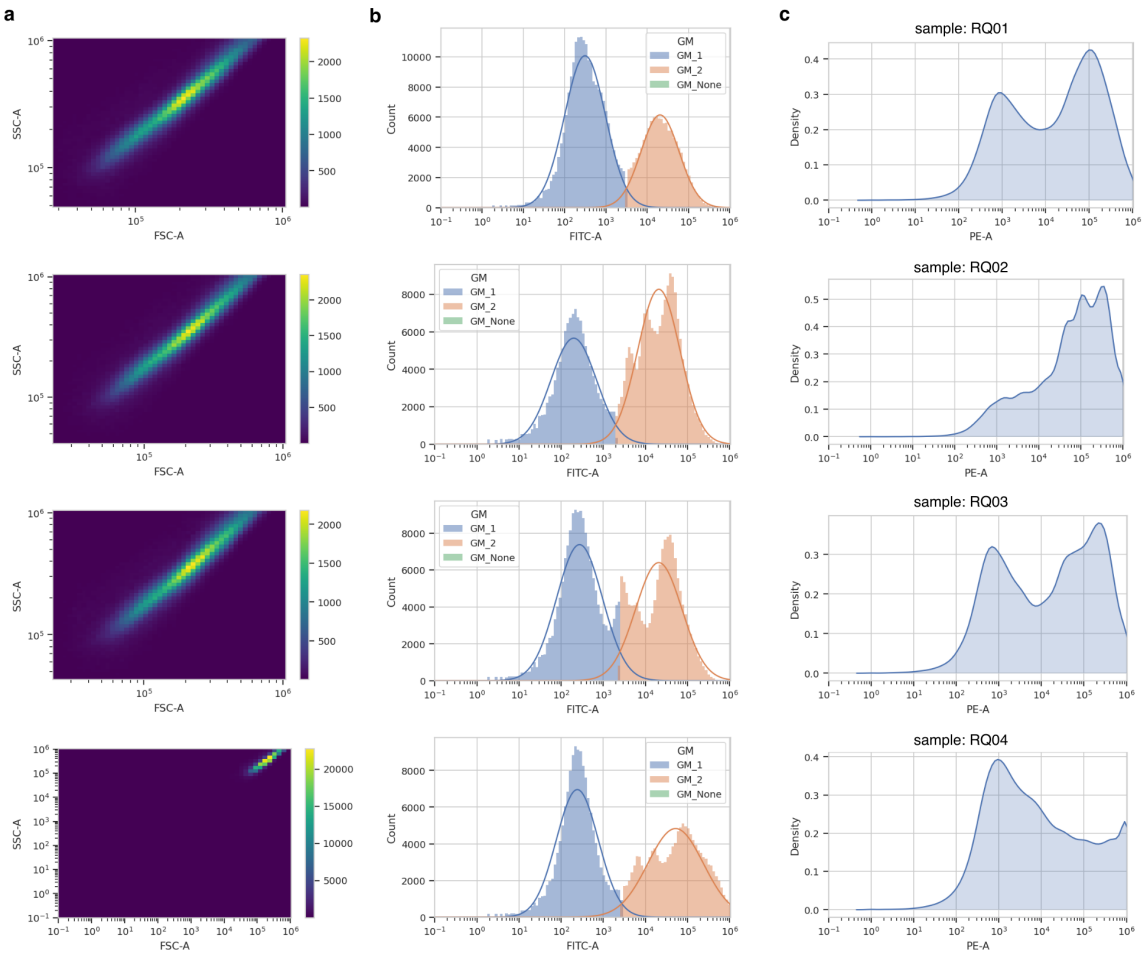

Figure 11 | Flow cytometry analysis of yeast display binding assays a, Forward- and side-scatter (FSC-A vs SSC-A) density plots of yeast displaying linearly fused DNA-binding proteins, with singlet populations automatically gated using Gaussian mixture modeling in CytoFlow. **b**, Two-component Gaussian fits of FITC-A fluorescence (anti-c-Myc–FITC) used to identify surface-expressing cells. **c**, PE-A fluorescence distributions of gated expressing populations for representative designs (RQ-01–RQ-04), showing streptavidin-PE detection of bound biotinylated dsDNA. Median PE/FITC ratios from these gated events were used to quantify apparent binding affinities in Fig. 10h.

5

Concluding Discussion

The de novo design of nucleic acid tertiary structures has been a longstanding challenge due to their intrinsic flexibility and the relatively small number of structures available for training deep learning models compared to proteins. Previous efforts, including adaptations of diffusion models for RNA, have been restricted to short sequences, exhibit limited structural diversity, and lack robust motif scaffolding or experimental validation, leaving the physical plausibility of their outputs uncertain [18, 19, 20, 21, 22] . RFDpoly enables user-guided generation of structured RNA with intricate, native-like folds, going beyond the regular origami-style topologies that have long defined the field of three-dimensional nucleic acid design. The use of randomly-sampled motif masks and conditional feature labels during training allows the model to learn local features of molecular structure, and to generalize these globally at inference time. Explicit hydrogen bond conditioning during RFDpoly's denoising process enables control over the topologies of the structures generated through the installation and enrichment of base-interaction networks. The successful design and cryo-EM validation of a 240-nucleotide RNA pseudocycle demonstrates that RFDpoly can produce novel RNAs with intricate folds. Despite being trained on a relatively small dataset of experimentally determined RNA structures, RFDpoly generalizes beyond known motifs (Extended Data Fig. 2a), in some cases producing designs more stable than their natural counterparts (Extended Data Fig. 5). The ability of RFDpoly to generate both base pair-conditioned and unconditional structures with high accuracy and diversity suggests that it does not simply memorize motifs from the training data, but

instead learns fundamental assembly principles, enabling the exploration and creation of physically viable, previously unobserved RNA folds.

The ability of RFDpoly to generate protein-nucleic acid assemblies enables the design of increasingly complex structures by reusing existing components. Combining the capabilities of RFDpoly to carry out polymer-class conditioning, 2D-templated scaffolding of preexisting motifs, and base pair templating to guide nucleic acid secondary structures enabled the design of the protein-DNA assemblies in Fig. 10. The close agreement of the crystal structure of a designed protein-DNA assembly, RQ-01, with the design model showcases the ability of RFDpoly to generate atomically accurate designs and to stabilize irregular (bent) DNA backbone structures. The relative placements of the input motif sets were not pre-defined within a global coordinate frame as in previous work: instead, the model inferred a physically plausible arrangement while generating the connecting backbones, eliminating the need for manual motif positioning, and enabling joint optimization of motif placement and backbone generation.

Despite the many new design capabilities described herein, there are still many avenues to explore to expand our ability to design RNA and nucleoprotein complexes. Developing compact, partner-specific protein-RNA interfaces would unlock fully RNA-centric scaffolds for hierarchical assembly, complementing the DBP-DNA fusions shown in this work. The development of more reliable tertiary structure prediction methods for protein-nucleic acid assemblies would considerably improve in silico screening of generated designs, and more broadly sharpen our ability to interpret the folding principles that underlie them [45] . While RFDpoly was not explicitly trained to model atomized ligands, future work incorporating greater chemical diversity (small-molecules, noncanonical nucleotides, and post-synthetic modifications) would further extend generative capabilities, and enable the design of aptamers and riboswitches programmed to bind more generalized targets. Finally, much of the utility of RNA in nature arises from conformational flexibility and stimulus-dependent behavior; fully realizing this potential in de novo design will require designing for functions of interest and pairing these efforts with experimental characterization of activity, responsiveness, and dynamics. Ultimately, the continued development of generative frameworks

like RFDpoly will likely transform the design of structured nucleic acids and proteins, unite protein design and DNA origami methods, and enable rapid, precise engineering of biomolecular assemblies for synthetic biology, therapeutic control, and beyond.

End Matter

Data Availability

Structure files for RNA pseudocycle pC2-240-04, and DBP-fusion RQ-01 have been deposited in the PDB, and will be publicly released at the time of publication.

Structure files for computational design models are included with this manuscript as supplementary data.

Experimental SHAPE-seq data and in silico scoring metrics for compact (240 nt) de novo RNA pseudocycles are provided as supplementary materials. Data for OpenKnot designs are available at: https://github.com/eternagame/OpenKnotAIDesignData.

Code Availability

The code for running the RFDpoly has been released as a GitHub repository, which includes links to download model weights: https://github.com/RosettaCommons/RFDpoly

Interactive tutorials highlighting the design techniques used in this paper are available on the RFDpoly GitHub page, and are also included as Supplementary Material files in this manuscript.

Acknowledgements

We thank the Eterna project and the Das lab (Stanford, HHMI) for coordination and experimental evaluation of the Eterna OpenKnot challenge.

The authors would like to thank Lance Stewart and Lynda Stuart for their work in maintaining operations at the Institute for Protein Design; Luki Goldschmidt and Kandise VanWormer for maintaining the computational and wet lab resources that were used in this research; Madison Kennedy and Bulat Faezov for technical help in preparation of this manuscript; Han Altae-Tran, Pooja D Bandawane, Andrew C. Hunt, Angela M. Yu, and Florian Praetorius for experimental advice and insightful discussions over the course of this project.

A

Methods

A.1 Model Development and Training

A.1.1 Motif templating

In RFDpoly, motif geometry is encoded as pairwise template features (2D track) and torsion vectors (1D track), while the 3D track receives only denoised frame coordinates and self-conditioning. This enables motif sets to be grouped and placed independently of a global coordinate frame. Both diffused and motif regions of training structures were subject to random perturbation of frame rotation and translation, since motif geometry was templated exclusively through RoseTTAfold's pair track using a two-dimensional motif masking scheme, as described previously [46]. Motif chunks were sampled as islands within input structures, with subsets randomly selected to have their relative distances and orientations locked or resampled, which emulated pairwise grouping of motif sets at inference time. While training the model to predict atom placement via torsion propagation was necessary for placing the many nucleic acid backbone atoms in diffused regions, even protein structure generation benefitted from proper torsion predictions in motif regions: our use of pairwise templating necessitated the use of motif torsion vectors to correctly reconstruct full sidechain atom sets within the coordinate frame of the newly generated structure. In previous versions of RFdiffusion, motifs remained fixed in their global frame, allowing sidechain and other

atoms to be directly replaced from input structures after diffusion. In our 2D motif templating, motif placements arise during denoising and thus cannot use sidechain replacement. Although backbone frames can be defined from inter-residue distances and orientations [47], full backbone and sidechain generation via torsion predictions required expanded training tasks, enabling the model to integrate distance maps, torsion templates, and sequence for full-atom structures.

A.1.2 Multimodal training

The standard diffusion task is to predict a ground-truth structure from a noised input, with the sequence masked in diffused regions and shown in motif regions. We added additional randomly sampled tasks: (i) predicting full sidechain coordinates from noised structure plus ground-truth sequence and torsions, and (ii) predicting backbone, sequence, and torsions from corrupted backbones to populate all atoms in both diffused and motif regions (the latter in a distinct global reference frame). This multimodal regime enhanced two design functions: (1) respecting full-atom 2D-templated motifs rather than just backbones, and (2) predicting sequences for generated structures, enabling autoregressive sequence-structure codesign during denoising. To handle the added complexity of nucleic acid backbones and 2D-templated motifs, we introduced auxiliary losses (beyond translation and rotation) to score torsion angle predictions and atom placement in local frames.

A.1.3 base pair feature encoding

For base pair conditioning, we encode a three-channel one-hot tensor, $A \in \{0,1\}^{L\times L\times 3}$ on the 2D track: [1,0,0] = unspecified (no constraint), [0,1,0] = explicitly non-pairing, and [0,0,1] = paired for nucleotide pair (i, j). Tensor A is symmetric, the diagonal is set to unspecified, and all entries default to unspecified unless base pair information is provided, and the 2D encoding permits definition of arbitrary-order pseudoknots. Per-nucleotide encoding was used instead of previously used block-adjacency conventions to indicate regional pairings, because block-level submatrices could not explicitly define the orientation of paired strands without an additional orientation feature. In

contrast, orientation can be explicitly defined for each paired region, as whether strands are oriented parallel or antiparallel is reflected by whether their corresponding submatrices are diagonal or the anti-diagonal, respectively. This fine-grained feature encoding provides explicit control over base pairing arrangements and enables the generation of irregular nucleic acid structures such as triple helices or G-quadruplexes.

A.1.4 Inference-time controls

A detailed outline of structural control methods, with many examples, covering multi-polymer design, 2D-templated motif scaffolding, and nucleic acid secondary structure conditioning, will be provided as code documentation and demonstrated through design tutorials, available at the time of publication.

A.2 Computational Design Protocols

A.2.1 RNA pseudocycle design

Base pair templates matrices were generated from coarse-grain topological specification, as shown in Fig. 6a. These templates were used to condition RNA structure generation using symmetrically propagated noise and pseudo-symmetric single-chain connectivity, as described previously [48].

A.2.2 Design of linearly-fused protein-DNA complexes

Three de novo DNA-binding proteins (DBP35opt, DBP48, DBP57) bound to their target DNAs (TGCACAT, GCCGC, ATCCAGA) [40] served as scaffolds for multi-chain protein–DNA fusions in parallel orientation (Fig. 10b, bottom). 2D motif templating enabled the relative placement of the complexes during denoising. A total of 6,000 diffused designs were generated. DNA linkers were specified at inference, while protein linkers were designed with LigandMPNN [25]. Structures were predicted with RoseTTAFoldNA (RFNA) [17] and filtered by pLDDT ≥ 0.87, backbone RMSD $\leq$

2.2 Å, and protein all-atom RMSD $\leq$ 3.5 Å (after alignment via all DNA atoms). Ninety-three designs passed these filters, including RQ01–RQ04 (Fig. 10h).

A.2.3 Design of semi-symmetric protein-DNA complexes

Semi-symmetric DBP fusions were designed in RFDpoly with symmetric noise, using motif scaffolding to place two DBP motifs per protein chain and their cognate motifs in DNA chains. Secondary structure conditioning was used to give DNA chains their characteristic strand-exchanging topology. An interfacial contact potential promoted protein-protein contacts in DBP fusion regions during denoising. Protein and DNA sequences were designed with NA-MPNN. Symmetric constraints guided protein sequence-design, while asymmetric sequence-design was used for DNA chains in order to produce DNA heterodimers or heterotrimers that would disfavor self-association. Some designs had their sequences further diversified by using ProteinMPNN [24] to redesign specific residues at protein-protein interfaces. Forty complexes were ordered and tested: 29 with disjoint protein domains and 11 with symmetric protein-protein interfaces.

A.2.4 Sequence design and sidechain generation

Sequence design could be performed using two methods. (1) RFDpoly-generated backbones could undergo inverse-folding sequence assignment using NA-MPNN [23]; PyRosetta was then used to thread MPNN sequences onto backbones, build sidechains, and perform quick repacking for reasonable placement. (2) Alternatively, RoseTTAfold's own sequence predictions could be used to autoregressively assign sequences during the denoising process, combining rigid-body and torsion parameters with sequence identity to produce full-atom models.

A.3 Experimental Methods

A.3.1 RNA transcription and purification

DNA templates were purchased from Twist and GenScript. In vitro transcription, purification, and refolding followed protocols described in previous work [49]. Transcription reactions used reagents from the NEB HiScribe™ reaction kit. The IVT products were purified with the QIAGEN RNeasy kit. Refolding was done in 50 mM Na-HEPES (pH 8.0) by heating to 90°C for 3 min and cooling for ≥10 min at room temperature. Samples were diluted to a final buffer of 10 mM $MgCl_2$. All RNAs carried a 5' GG leader to promote transcription, adding 2 nt to each synthesized monomer.

A.3.2 Eterna OpenKnot challenges

Puzzle base pair templates were given by Eterna OpenKnot puzzles in dot-bracket notation. These strings were fed to RFDpoly to produce backbones that could then be designed by NA-MPNN[23]. Sequences were filtered for in silico self-consistency using both RibonanzaNet-1D and -2D, due to its reported state-of-the-art accuracy in predicting base pair interactions, relative to alternative RNA secondary prediction oracles[28]. Sequences that passed RNet filtering were subjected to a second round of self-consistency filtering by predicting full-atom tertiary structures (using Chai-1 [30] for Round 6 and AlphaFold 3[31] for Rounds 7a and 7b), from which base pairs were extracted and again used to compute predicted F_1-scores. The OpenKnot organizers experimentally assessed the secondary structures of designed sequences using SHAPE-seq[35, 36, 37].

Additional information about the Eterna OpenKnot challenge (including puzzles and secondary structures) is available at: https://eternagame.org/challenges/11843006

A.3.3 Yeast expression of linearly fused DNA-binding proteins

DBP fusion sequences were optimized for S. cerevisiae, synthesized as IDT E-blocks with pETCON3 (Addgene #45121) adaptors, and cloned/transformed by in vivo homologous recombination. Specif-

ically, linearized pETCON3 vector was mixed with LiOA (5.5 ng/µL vector, 0.67 M LiOA). 3 µL of pETCON3–LiOA solution and 5 µL of 10 ng/µL E-block inserts were added to 96-well PCR plates and incubated for 5 minutes at room temperature. 30 µL PEG–LiOA solution (43.3% PEG 3,350, 0.13 M LiOA) and 10 µL competent EBY100 yeast were added. Mixtures were incubated in a BioRad T100 thermal cycler: 30 minutes at 30°C, then 20 minutes at 42°C. Transformations were briefly centrifuged (accelerated to 4,000g, then immediately decelerated without braking). Pellets were washed by resuspension in 50 µL water and re-pelleting. Cells were resuspended in 200 µL C-Trp-Ura medium with 2% (w/v) glucose and transferred to 96-well culture plates. Cultures were incubated for ˜40 hours at 30°C with shaking. Cultures were either stored (1:1 with 50% glycerol at -80°C) or induced by 8-fold dilution into SGCAA medium with 0.2% glucose and incubation at 30°C for 16–24 hours with shaking.

A.3.4 Yeast display binding assays

20 µL of SGCAA cultures were prepared as described above, and distributed into Corning® 96-well V-bottom plates. Cells were washed with PBSF (PBS + 1% BSA), and then resuspended in 30 µL of 1 µM biotinylated dsDNA and incubated for 30 minutes at room temperature with shaking. After PBSF washing, cells were stained in 30 µL solution (32 µg/mL anti-c-Myc-FITC, ICL Lab; 32 µg/mL streptavidin-PE, Thermo Fisher, in PBSF) for 20 minutes at room temperature with shaking. Cells were then washed and resuspended in 200 µL PBSF. Binding was assessed by flow cytometry on an Attune NxT with an autosampler. Data were analyzed with custom Python code and CytoFlow, as in Glasscock et al. [40]. Expression gating used the CytoFlow Gaussian mixture model, and binding was quantified as the PE/FITC intensity ratio for gated events. Streptavidin-PE was used to detect biotinylated dsDNA and anti-c-Myc-FITC to detect C-terminal Myc tags on surface-displayed proteins.

A.3.5 Bacterial expression and purification of DBP fusions

DBP fusion sequences were codon-optimized for E. coli, synthesized as G-blocks, and cloned into plasmid LM627 (Addgene #191551) by Golden Gate. Plasmids were transformed into BL21(DE3) E. coli. Transformants were inoculated into 2–10 mL LB + 50 mg/L kanamycin starter cultures and incubated overnight at 37°C, 225 rpm. Starter cultures were diluted 1:50 into 50 mL or 1 L LB + kanamycin and grown at 37°C, 225 rpm to OD600 0.6–0.8. Expression was induced with 1 mM IPTG, and cultures were grown for ˜16 hours at 18°C, 225 rpm. Cells were harvested by centrifugation at 3,000g for 15 minutes. Pellets were resuspended in high-salt lysis buffer (2 M NaCl, 20 mM Tris-HCl, EDTA-free protease inhibitor) and sonicated (QSonica Q500, 4-pronged horn, 5 minutes, 70%). Lysates were clarified (14,000g, 30 minutes) and supernatants were passed over 0.5-1.0 mL Ni-NTA resin in gravity columns. Resin was washed with 20 Column Volumes (CV) of high-salt buffer (2 M NaCl, 20 mM Tris-HCl, 30 mM imidazole, pH 8.0). Proteins were eluted with 2 CV high-salt buffer (2 M NaCl, 20 mM Tris-HCl, 300 mM imidazole) or, for crystallography, cleaved on-column to remove the SNAC-His tag. For cleavage, columns were washed with 5 CV SNAC buffer (100 mM CHES, 100 mM acetone oxime, 100 mM NaCl, 500 mM GnCl, pH 8.6), incubated overnight at room temperature in 5 CV buffer + 0.2 mM $NiCl_2$, and flowthrough collected. Cleaved or uncleaved eluates were concentrated to 1 mL, filtered, and injected on a Superdex S75 Increase 10/300 GL column (ÄKTA Pure) at room temperature in SEC buffer. High-salt SEC buffer (2 M NaCl, 20 mM Tris-HCl, pH 8.0) was used for crystallography samples and mid-salt buffer (500 mM NaCl, 20 mM Tris-HCl, pH 8.0) for others. Monodisperse fractions were pooled, concentrated with 3 kDa cutoff spin filters (Amicon, Millipore Sigma), and stored at 4°C. Protein concentrations were determined by A280 on a NanoDrop spectrometer (Thermo Fisher Scientific).

A.3.6 Bacterial expression and purification of DBP fusions

Single-stranded DNA (ssDNA; 97 nt) was purchased from Integrated DNA Technologies (IDT). Equimolar strands were mixed, heated to 72°C, and slow-cooled to 12°C over 23 hours to form

strand-exchanging heterodimers or heterotrimers. Annealed complexes were size-purified by HPLC, collecting fractions at expected UV peaks. Purified DNA complexes were mixed with their associated DBP-fusion proteins to form final protein-DNA complexes, which were again resolved by HPLC to isolate full assemblies, which were then characterized using nsEM.

A.3.7 nsEM characterization

For each sample (RNA pseudocycles and protein-DNA complexes), glow-discharged Lacey Carbon copper grids (1 µm holes, 5 µm spacing) received 4 µL diluted RNA, incubated 20 seconds, blotted, then rinsed with 4 µL nuclease-free H_2O and blotted. Grids were stained three times with 4 µL of 2% uranyl formate, each for 10 seconds before blotting. Grids dried for $\geq$5 minutes before loading into a Talos L120C microscope. Imaging used 120 kV, 2.7 mm Cs, and a Ceta 4K CCD (4096×4096). Automated collection was done with EPU (Thermo Fisher). Data were processed in cryoSPARC to create 2D class averages and 3D reconstructed volumes [50].

A.4 Cryo-EM sample preparation

A.4.1 Cryo-EM sample preparation

2 µL of purified RNA at a concentration of 43 mg/mL in 10 mM $MgCl_2$ was applied to glow-discharged 400-mesh 2/2 c-flat holey carbon grids (EMS CFT-223C-100). Grids were vitrified on a Vitrobot Mark IV, at 22°C with 100% humidity for all. Blotting was done using a 6.5 - 7.5 second blot time, a blot force of 0, and a 7.5 second wait time before being immediately plunge frozen into liquid ethane. Grids were clipped and kept in liquid nitrogen until loading.

A.4.2 Cryo-EM data collection and processing

For RNA pseudocycle pC2-240-04, data were collected on a 300 kV FEI Titan Krios. Movies were acquired in SerialEM using beam shifts, at 0.83 Å/pixel. Data were processed in CryoSPARC v4.7[50]. Iterative rounds of particle picking, extraction (400 px box size), and 2D classification were

performed. Iterative rounds of ab initio reconstruction were performed (3-15 classes per round), where particle sets associated with volumes of the correct size and shape were selected and fed into subsequent rounds of ab initio reconstruction. Once helical pitch and hairpin extensions became apparent, heterogeneous refinement was performed to remove junk particles, while the target class and its particles were then fed into non-uniform refinement.

A.4.3 Cryo-EM model building

The design model was rigid-body docked into the final cryo-EM map in ChimeraX, and backbone-refined to fit the experimental density using ISOLDE[39]. Final RNA structure models were built by iterative map-fitting MD and partial diffusion, where initial fits from ISOLDE were refined by running RFDpoly partial diffusion/denoising to yield cleaner backbone geometry while introducing subtle structural diversification. Five iterations of this process were performed, followed by quick PyRosetta refinement, to produce the final experimentally refined model.

A.4.4 Comparison of Cryo-EM model against native RNA folds

To assess whether the designed RNA pseudocycle pC2-240-04 adopts a fold similar to known RNAs, we compared its experimentally-built cryo-EM model against all representative RNA structures in the RNAsolo database[51] (4,089 structures). Structural alignments were performed using USalign[34] in RNA mode, with pC2-240-04 as the aligned structure and native RNAs as the reference targets. Candidate matches were ranked by TM-score normalized to the design length, and only alignments with $\geq$0.5 coverage (coverage = $L_{\text{aligned}}/L_{\text{design}}$) were considered. The closest structural match was chain-A of the plant mitochondrial ribosome (PDB 6XYW; $L_{\text{aligned}} = 133$, $L_{\text{design}} = 240$, $L_{\text{native}} = 2,842$), which achieved a TM-score of 0.329, supporting the novelty of our de novo designed RNA fold.

B

Motif scaffolding controls

This section accompanies `demo01_general_design.ipynb` (Section 3, *Hierarchical design and 2D motif templating*) and provides additional explanation for pairwise motif-templating design controls, and how RFDpoly can be used to resample motif docks and relative-positions during inference.

B.1 Overview: hierarchical motif templating with controlled rigid-body constraints

RFDpoly motif scaffolding supports hierarchical control over which parts of an input motif set remain rigidly constrained during inference. At a minimum, the method templates *intra-motif* geometry (i.e., preserves the internal coordinates within each provided motif fragment). In addition, users can optionally template selected *inter-motif* relationships so that specific motif groups retain their relative placement (for example, a protein bound to a DNA segment), while allowing other motif groups to move relative to each other in order to explore alternative global arrangements.

We illustrate this control using an input structure containing two DNA-binding proteins (DBPs), each bound to a short DNA duplex. The design goal is to place both DBPs on a single continuous DNA helix while (i) preserving each protein–DNA binding interface, (ii) inpainting missing DNA to connect the binding sites (e.g., inserting a spacer), (iii) allowing the two protein–DNA units to move relative to each other, and (iv) optionally extending DNA upstream/downstream to form a

longer helix (Supplementary Fig. S1).

B.2 Contig specification and motif identifiers

Contigs ("contiguous regions") define the topology of the output structure by specifying which motif fragments are fixed and where diffused (inpainted) regions are inserted. For the purpose of controlling which inter-motif template relationships are exposed during inference, each motif fragment appearing in the contig specification is assigned a motif identifier (lowercase letters) based on the the order it occurs in the full set of contig strings (Supplementary Fig. S1b). These motif identifiers are distinct from the chain IDs in the input PDB (typically uppercase), which avoids ambiguity when a single input chain contributes multiple fragments or when multiple input chains are fused into a single output chain.

B.3 `inference.ij_visible`: selectively templating inter-motif geometry

The argument `inference.ij_visible` controls which motif fragments "see" each other's inter-motif template features during inference, and therefore which inter-motif distances and orientations are constrained (templated) versus masked.

Motifs listed within the same group (i.e., within the same substring) have their pairwise inter-motif template features enabled (*visible*), which preserves their relative placement up to the model's templated representation. Motifs in different groups (separated by `-`) have their off-diagonal inter-motif template features masked (*invisible*), allowing those groups to move relative to one another during inference (Supplementary Fig. S1c).

For example, the `ij_visible` string

```
acf-bde
```

specifies two independent groups, $\{a, c, f\}$ and $\{b, d, e\}$. Pairwise relationships are templated within

each group, but not between groups. In the DBP example, this allows rigid preservation of each protein–DNA binding pose within a DBP–DNA unit while permitting the two units to explore different relative placements along a shared DNA helix (Supplementary Fig. S1b–c, top).

B.4 Example: placing two DBPs on a single DNA helix by DNA inpainting

In the first configuration, the two protein motifs are kept as separate output chains while the DNA is fused into a continuous helix. Each DBP is constrained to remain in its bound pose relative to its local DNA motif via `ij_visible` grouping, while the relative arrangement between the two DBP–DNA units is left unconstrained. Missing DNA between binding sites is generated by inserting an inpainted spacer (e.g., 4 nt), and optional upstream/downstream DNA can be generated to extend the helix. Across independent diffusion trajectories, this setup preserves both binding interfaces while sampling global arrangements and subtle conformational variation in the intervening DNA (Supplementary Fig. S1a–c, top).

B.5 Alternative topology: protein fusion while preserving hierarchical constraints

In a second configuration, the two DBP motifs are connected into a single output chain by introducing a diffused (inpainted) protein linker, while still using the same underlying motif set. This enables exploration of designs in which the protein domains are physically linked, while the hierarchical templating logic remains unchanged: intra-motif geometry is preserved, selected inter-motif relationships can be templated within motif groups, and other degrees of freedom are left free for the model to generate (Supplementary Fig. S1b–c, bottom).

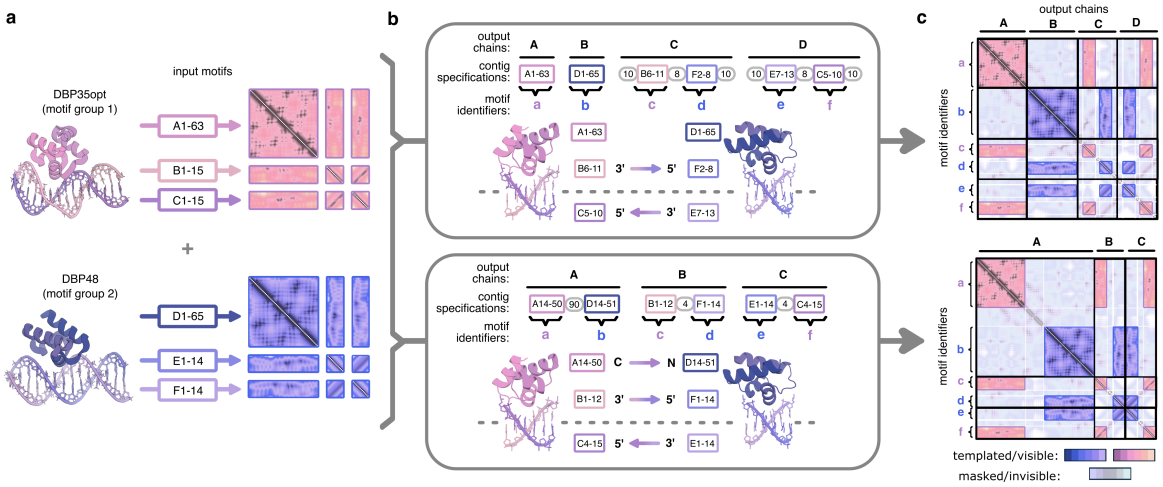

Figure S1 | Hierarchical design and 2D motif templating a, Input motif chains and contig specification, showing which intra- and inter-chain template distances are passed to the model. **b**, Contig layout and template control. Motif fragments specified in the contigs are assigned lowercase motif identifiers (in order of appearance). The `inference.ij_visible` argument specifies which motif identifiers have their inter-motif template relationships enabled ("visible") during inference. Motifs listed within the same group (e.g., `acf`) retain templated inter-motif geometry, whereas motifs separated by `-` (e.g., `acf-bde`) are mutually "invisible," allowing those groups to move relative to each other. Top: two DBP motifs are positioned along a shared DNA helix by fusing DNA while keeping DBP chains separate. Bottom: a modified topology in which the two DBP motifs are additionally connected into a single fused chain. **c**, Constructed 2D template (pairwise) features for the two-DBP docking arrangement (top) and two-DBP fusion arrangement (bottom). Intra-motif geometry is templated for all motifs, and selected inter-motif features are templated within `ij_visible` groups (colored), while off-diagonal inter-group features are masked (grey) and generated during inference.

C

Partial diffusion and multi-state design

This section accompanies the notebook `demo02_ensemble_modeling.ipynb`, which demonstrates an ensemble-aware RNA design workflow combining motif scaffolding, partial diffusion for conformational sampling, and multi-state inverse folding using tied NA-MPNN.

C.1 Motif preparation and aptamer scaffolding designs

As a model system, we used an AMP-binding RNA aptamer (PDB ID: 1AM0) as a fixed functional motif for scaffolding. Sequence and structural information were extracted from the deposited structure, and missing RNA regions were completed using AlphaFold 3 (AF3). The resulting structure file was then manually edited so that the AMP small molecule could be represented using standard biopolymer-style `ATOM` records, enabling it to be treated as a single-token, RNA-chain-like motif by RFDpoly.

To test how the extent of de novo scaffold influences stability and designability, we generated three *aptamer-scaffolding RNA* configurations (ASR1–ASR3) that vary the length of the diffused (non-motif) regions used to connect and buttress the fixed aptamer and AMP motif components (Supplementary Fig. S2a). The inference contig specifications were:

- **ASR1:** `contigmap.contigs=['29,A1-21,46,A30-40,10 B1-1']`
- **ASR2:** `contigmap.contigs=['27,A1-21,52,A30-40,8 B1-1']`

- **ASR3:** `contigmap.contigs=['29,A1-21,48,A30-40,8 B1-1']`

For each ASR setting, sequence–structure codesign was performed during the denoising trajectories, in order to yield both initial backbones and initial sequences for subsequent steps.

C.2 Ensemble generation by partial diffusion

RNA molecules often populate ensembles of nearby tertiary conformations. To explicitly sample plausible backbone variability without perturbing the binding-site geometry, we generated discrete conformational ensembles using partial diffusion starting from the scaffolded designs. In this procedure, the full sequence is held fixed, and only a subset of denoising steps is used to introduce limited noise and re-denoise, thereby sampling near-native fluctuations in the non-motif scaffold while preserving the aptamer/AMP motif geometry (Supplementary Fig. S2c).

For each ASR setting, we generated 10 ensemble members by running 10 independent partial-noise/denoise trajectories with `partial_T = 20` steps, parameterized relative to a full diffusion horizon of $T = 50$. Unless noted otherwise, ensemble visualizations overlay conformers after alignment on the fixed motif (binding-domain) region to emphasize sampled variation in the non-motif scaffold.

Note (*Optional extension – larger motions*): If a user wanted to sample larger-amplitude conformational changes without disrupting base pairing networks, `partial_T` can be increased and paired with secondary-structure constraints at inference time. Conceptually, this is particularly relevant for modeling conformational motions of riboswitch-like RNAs that exhibit multiple tertiary states while sharing a common secondary structure.

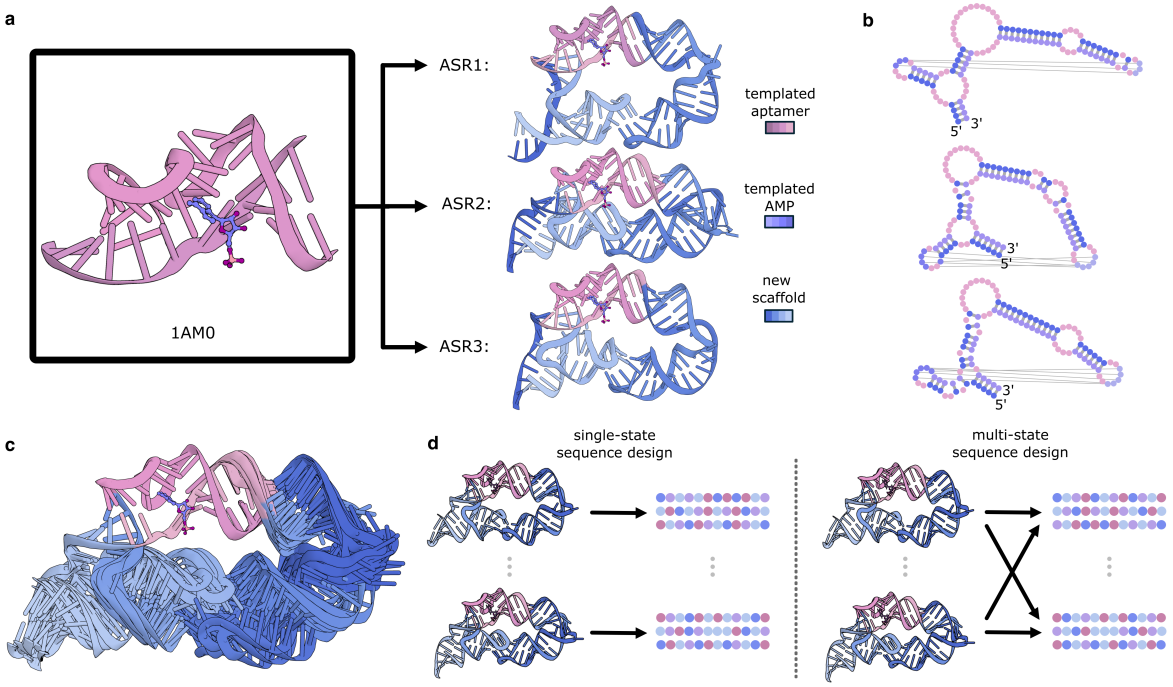

Figure S2 | Design strategies for scaffolding aptamer domains a, Design of three *aptamer-scaffolding RNA* configurations (ASR1–ASR3) to stabilize the binding domain of the AMP aptamer (PDB ID: 1AM0), showing the fixed (templated) aptamer/ligand motif and the newly generated scaffold. **b**, Secondary-structure representations for representative unconditioned diffusion samples of the three ASR topologies. **c**, Example conformational ensemble generated by partial diffusion from an ASR design, visualized as an overlay of ensemble members after alignment on the fixed binding-domain motif. **d**, Schematic comparison of single-state versus multi-state (tied) NA-MPNN sequence design on partially diffused ensembles.

C.3 Single-state versus multi-state (tied) NA-MPNN sequence design

We compared two ways of leveraging the partially diffused conformational ensemble for inverse folding with NA-MPNN (Supplementary Fig. S2d). In both cases, the AMP/aptamer motif geometry is held fixed and the design objective is to identify sequences that fold reliably into scaffolded states consistent with the generated ensemble.

C.3.1 Single-state design across an ensemble (backbone-diversified inputs)

Each partially diffused conformer was treated as an independent single-structure inverse-folding target (an "ensemble member"). For each ensemble member, we sampled 5 sequences with NA-MPNN, yielding 50 sequences per ASR condition (10 backbones $\times$ 5 sequences/backbone). This baseline isolates the effect of backbone diversification: it tests whether providing NA-MPNN with a set of closely related backbone targets (while maintaining the same fixed motif geometry) improves the likelihood of discovering sequences that are robust to small structural perturbations in the non-motif scaffold.

C.3.2 Multi-state design with tied NA-MPNN (one sequence optimized across conformers)

In multi-state mode, corresponding residue positions are *tied* across ensemble members, such that NA-MPNN is tasked with identifying sequences that are jointly compatible with the full set of conformers (i.e., ensemble-aware inverse folding). Practically, we constructed a combined multi-structure input containing all 10 ensemble members and translated each conformer by a large offset (900 Å) to ensure spatial separation. This translation is used solely to prevent cross-conformer geometric neighbor edges and therefore eliminate inter-structure message passing through the NA-MPNN structure graph, while maintaining coupling of sequence decisions across conformers through the tied-residue constraint during autoregressive decoding. The translation strategy is designed to

yield disconnected structure graphs across conformers. Let d denote the typical maximum spatial extent of a single conformer (e.g., the largest atom–atom distance within one structure), and let D denote the minimum inter-conformer separation in the combined input (i.e., the smallest distance between any atom in conformer b and any atom in conformer b'). Choosing translations such that $D \gg d$ ensures that, under standard local neighbor constructions (including k-nearest-neighbor graphs), each node's nearest neighbors lie within the same conformer, and thus no geometric edges connect distinct ensemble members. Under this construction, conformers remain coupled only through the tied-residue constraint at the sequence level during decoding. For each multi-state NA-MPNN run (10 conformers jointly), we sampled 50 sequences, producing 50 sequences per ASR condition—matched to the single-state baseline for a controlled comparison.

C.4 In silico evaluation by AlphaFold 3 self-consistency

To compare the single-state and multi-state sequence-design regimes, we predicted structures for each designed sequence using AlphaFold 3 (AF3) and evaluated predicted self-consistency relative to the design backbone(s) (Supplementary Fig. S3a,c). Each point corresponds to one designed sequence, summarized by averaging across two sources of variability: (i) multiple AF3 predictions per sequence ($M = 5$ independent AF3 prediction trajectories), and (ii) one or more reference backbones (B) used for scoring. For single-state designs, $B = 1$ (the specific backbone on which that sequence was designed). For multi-state designs, $B = 10$ (all ensemble members), reflecting the explicit objective that a single sequence should be compatible with the ensemble.

C.4.1 AF3 self-consistency aggregation (informal)

For any metric q computed between a predicted structure and a reference backbone (e.g., TM-score, lDDT, secondary-structure F1), we define a per-sequence summary score as the mean of q across all AF3 samples and all reference backbones: $\bar{q}(s) \approx \mathrm{mean}_{m\in\{1,\dots,M\},\, b\in\{1,\dots,B\}}\, q\Big(\hat{\mathbf{X}}^{(m)}(s),\, \mathbf{X}^{(b)}\Big)$. This aggregation yields a single comparable score per designed sequence while reducing sensitivity

to AF3 stochasticity and (for multi-state design) explicitly reflecting agreement across the intended ensemble of backbone states.

C.5 *In silico* performance of AMP–aptamer scaffold design strategies

Following NA-MPNN sequence design, AF3 evaluation indicated that sequences designed in the single-state regime outperformed multi-state (tied) designs in overall fold self-consistency when assessed by full-structure agreement between design models and AF3 predictions (lDDT, TM-score, and secondary-structure F1; Supplementary Fig. S3a). These metrics enabled straightforward *in silico* filtering of design–prediction pairs and highlighted a subset of candidates with remarkably close tertiary agreement to their intended scaffolded states (Supplementary Fig. S3b).

Consistent with this trend, designs ranked by ligand- and interface-focused metrics—including interfacial confidence (iPTM, interchain min-pAE) and ligand RMSD—were also enriched among single-state designs relative to multi-state designs (Supplementary Fig. S3c). Among the top-ranked candidates, AF3 predictions frequently captured the AMP-binding geometry with high precision, including accurate ligand placement and well-formed local binding-site interactions (Supplementary Fig. S3d).

Taken together, these results suggest that, for the AMP-aptamer scaffolding benchmark studied here, single-state NA-MPNN sequence design provides stronger *in silico* performance than multi-state (tied) sequence design when evaluated by AF3 self-consistency and ligand-interface criteria. At the same time, the partial-diffusion ensemble provides a useful and general mechanism to diversify non-motif scaffold geometries while preserving a fixed functional motif, enabling systematic comparisons of sequence-design strategies under matched sampling budgets. More broadly, this workflow supports a design hypothesis that is especially relevant for ligand-binding RNAs: sampling near-native conformational variability around a binding-ready motif and then selecting sequences that remain self-consistent under *in silico* folding can help identify scaffolds that sta-

bilize and preorganize the ligand-binding conformation, potentially reducing the entropic cost of binding-associated conformational restriction.

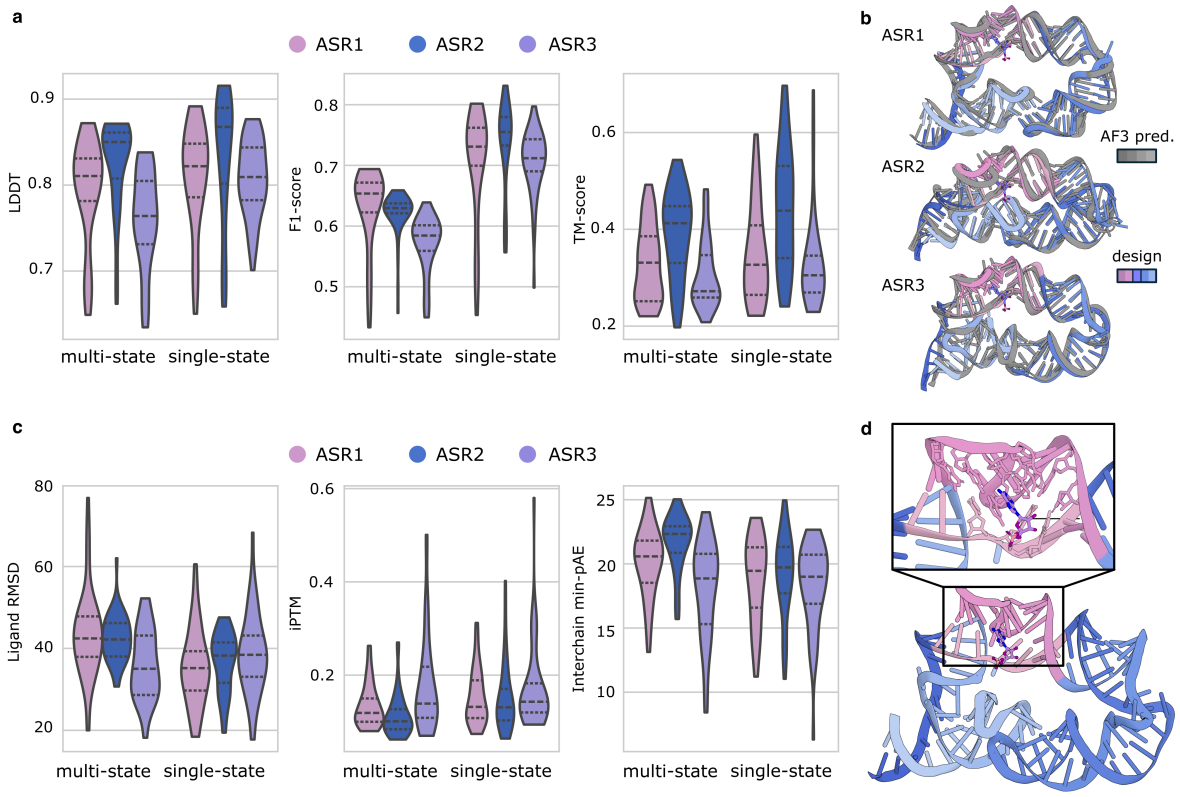

Figure S3 | In silico predictions for ASR designs a, Full-structure self-consistency metrics comparing AF3 predictions to the corresponding ASR design models (lDDT, TM-score, and secondary-structure F1), stratified by multi-state versus single-state NA-MPNN design. **b**, Representative AF3 predictions for selected ASR designs shown as overlays of AF3-predicted structures and their corresponding design models. **c**, Ligand- and interface-focused *in silico* metrics for ASR design campaigns, including ligand RMSD and interface confidence measures (iPTM and interchain min-pAE), comparing multi-state versus single-state design. **d**, Example AF3 prediction for an ASR3 design (single-state NA-MPNN), highlighting precise placement of the AMP ligand in the binding site.

D

Constructing base pair templates at inference time

RFDpoly supports explicit *secondary-structure conditioning* during diffusion by converting user-specified base pair constraints into a *base pair template* (a sparse pairing graph over output residue indices). This template is constructed once at the start of inference and is then used as conditioning throughout the denoising trajectory.

D.1 Overview and visualization conventions

Supplementary Fig. S4 illustrates the conditioning tensors used in several common settings. In each column, the top panel shows an example design output, and the bottom panel shows the corresponding conditional features tensor: **mask** (dark), **pairs** (magenta), and **loops** (blue). Fully unconditional generation corresponds to a fully masked tensor (**Supplementary Fig. S4a**). Secondary-structure-conditioned generation corresponds to enabling pair and loop features at user-specified positions (**Supplementary Fig. S4b–d**).

D.2 Input formats

base pair templates can be specified in two equivalent ways:

(i) Dot–bracket-style strings. For single-chain designs (or per-chain segments), users may provide a secondary-structure string using a dot–bracket-like alphabet:

- paired symbols: `5/3` (analogous to `(/)`), with additional matched symbol pairs allowed for pseudoknots;
- `.` explicitly unpaired positions;
- `?` unspecified positions (left unconstrained).

Because parentheses can be inconvenient in some command-line/Hydra settings, matched symbol pairs such as `5/3` are used in practice (see GitHub repository for full symbol documentation).

To place motifs at specific locations, users may provide a list of segment-scoped strings via `scaffoldguided.target_ss_string_list`, where each entry specifies an *output* chain and index range:

```
'<CHAIN_ID><START>-<END>:<secondary-structure-string>'
```

All indices refer to positions in the *generated output*. This form is convenient for specifying complex secondary-structure graphs, including pseudoknots (**Supplementary Fig. S4b**), and for explicitly marking loop regions that should remain unpaired.

(ii) Paired range lists. For multi-chain systems (or when specifying pairing at a higher level), users may provide explicit paired segments using `scaffoldguided.target_ss_pairs`, a list of range pairs:

```
['A1-20,B1-20', 'A21-40,C21-40', ...]
```

Each pair must have equal length and defines a helix-like set of base pairs between the two ranges. This form is useful for specifying full helical domains (**Supplementary Fig. S4c**) without writing a full secondary-structure string.

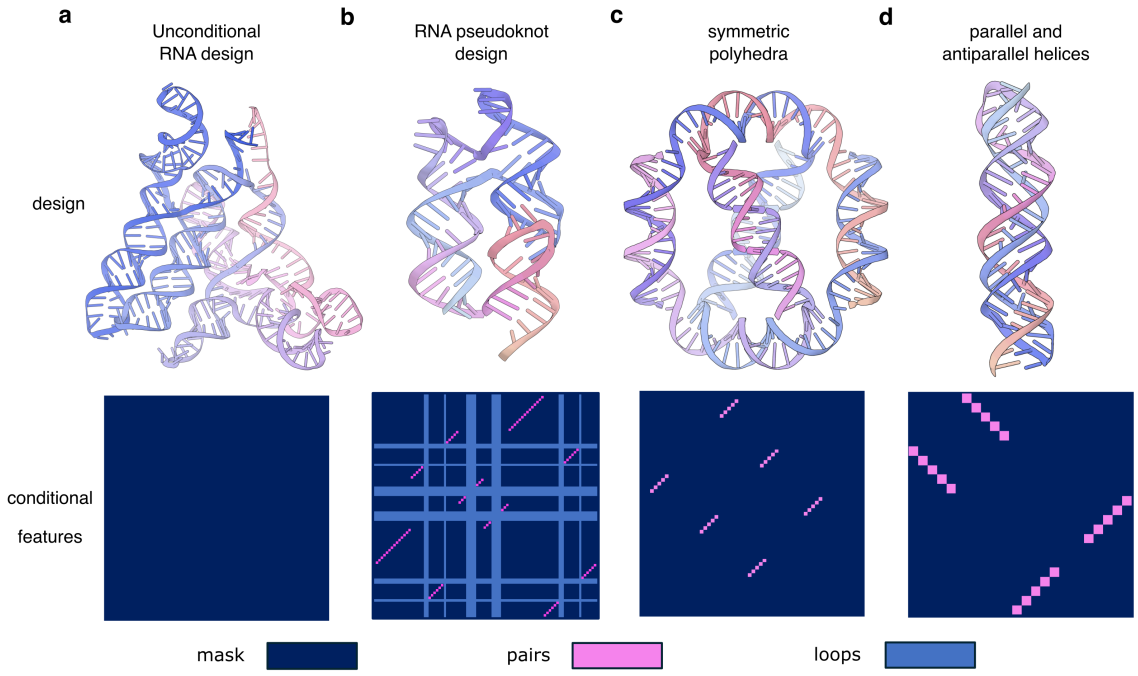

Figure S4 | Secondary-structure control with base pair templates Top: representative RFDpoly design outputs. Bottom: corresponding conditional-feature tensors used for secondary-structure conditioning at inference time, visualized as a residue–residue map with **mask** (dark; unconstrained), **pairs** (magenta; specified base pairs), and **loops** (blue; explicitly unpaired positions). **a,** Fully unconditional generation uses a fully masked tensor. **b,** Pseudoknot-conditioned design specified via segment-scoped secondary-structure strings (e.g., `scaffoldguided.target_ss_string_list`), which can encode both paired and explicitly unpaired positions. **c,** Helical domains specified by paired index-range lists (e.g., `scaffoldguided.target_ss_pairs`), which define the full extents of strand regions that should pair. **d,** Mixed parallel and antiparallel contacts enabled by providing paired index ranges together with explicit strand-orientation annotations (e.g., `scaffoldguided.target_ss_pair_ori`); antiparallel pairing is assumed by default, while parallel pairing is supported to condition non-standard tertiary motifs such as triplex-like architectures.

D.3 Template construction

All input formats are converted into a consistent internal representation: a set of residue–residue pairing edges $\mathcal{E} = \{(i, j)\}$ defined over output residue indices.

Parsing dot–bracket strings. For each matched symbol pair $(s_{\text{open}}, s_{\text{close}})$, the string is parsed with a stack: when s_{open} is encountered at position i, push i; when s_{close} is encountered at position j, pop the most recent i and add the edge (i, j) to $\mathcal{E}$. Positions marked `.` may be recorded as explicitly unpaired (loop constraints); positions marked `?` are left unconstrained. The resulting edge set produces sparse "pair" entries in the conditioning tensor (**Supplementary Fig. S4b**).

Expanding paired ranges. For each paired range specification $(a_1 : a_2,\ b_1 : b_2)$ with length L, the template adds L base pair edges according to an orientation rule (below). This generates contiguous diagonals of "pair" entries corresponding to full helical segments (**Supplementary Fig. S4c,d**).

D.4 Strand orientation

By default, paired segments are treated as *antiparallel*, consistent with canonical duplex geometry. For cases such as triplexes or noncanonical alignments, users may override the orientation per pair group via `scaffoldguided.target_ss_pair_ori`, a list with entries in {`A`, `P`} indicating antiparallel or parallel orientation, respectively.

Let $a_k = a_1 + k$ and $b_k = b_1 + k$ for $k = 0, \ldots, L - 1$. Then:

$$\text{antiparallel (A):} \quad (a_k,\ b_{L-1-k}) \in \mathcal{E}, \tag{D.1}$$

$$\text{parallel (P):} \quad (a_k,\ b_k) \in \mathcal{E}. \tag{D.2}$$

This convention enables simultaneous specification of antiparallel helices and parallel-strand contacts when constructing triplex-like architectures (**Supplementary Fig. S4d**).

D.5 Beyond pairwise constraints: multi-base contacts and forced loops

Some tertiary contacts cannot be expressed as disjoint base pairs (e.g., stapled loop–loop junctions or multiway contacts). RFDpoly therefore supports additional constraint lists that are converted into template constraints alongside $\mathcal{E}$:

- `scaffoldguided.force_multi_contacts`: a list of residue sets that are encouraged to co-localize, enabling distal loop contacts beyond simple pairing.
- `scaffoldguided.force_loops_list`: a list of ranges that are explicitly treated as loop/unpaired segments, used to prevent unintended helix formation and to stabilize intended junction geometry.

The resulting base pair templates are supplied to the inference engine as a fixed conditioning signal throughout denoising. Fully unconditional generation corresponds to applying only the mask (**Supplementary Fig. S4a**), whereas conditional generation introduces pair and loop features at specified residue indices (**Supplementary Fig. S4b–d**), constraining secondary structure while allowing the model to explore compatible 3D realizations.

D.6 Use during diffusion

In practice, these secondary-structure features should be interpreted as guidance rather than absolute constraints: during denoising, the model must continuously reconcile the user-provided 2D pairing template with the local and global geometric regularities it has learned from training data. The resulting structures therefore reflect a tradeoff between template adherence and biophysical feasibility – particularly at junctions, long-range contacts, or highly strained pairings – where enforcing an exact pairing graph could otherwise require implausible backbone geometry. This is consistent with the overall design interface in which base pair constraints are constructed before inference, and consistently applied as conditioning throughout the denoising trajectories.

To empirically assess how strongly conditioning biases the final pairing patterns, we evaluated secondary-structure agreement between target templates and diffusion outputs for all OpenKnot puzzles **Supplementary Fig. S5**. For these runs, we used sequence–structure codesign so that trajectories terminate with full-atom nucleic-acid representations (including base identities/coordinates), enabling direct extraction of output base pairs at the end of denoising without requiring a separate NA-MPNN design-and-repack step for evaluation. Across OpenKnot rounds 6, 7a, and 7b, we observe consistently high template agreement, and notably strong adherence even for the longest Round 7b targets (up to 240 nt).

Consistent with the role of the template as a conditioning signal (rather than a hard constraint), outputs reflect a balance between user-specified 2D patterns, 3D structural feasibility learned from training data, and the effect of additional stochastic noise perturbations applied throughout denoising trajectories; accordingly, some trajectories deviate locally from the exact target pairing graph even when the global fold is compatible. Nonetheless, conditioning produces high agreement between generated pairing patterns and the specified templates. These results support the practical utility of base pair conditioning as a robust mechanism for steering diffusion toward desired secondary-structure topologies, indicating that RFDpoly reliably follows complex secondary-structure specifications while retaining sufficient flexibility to realize physically plausible 3D folds.

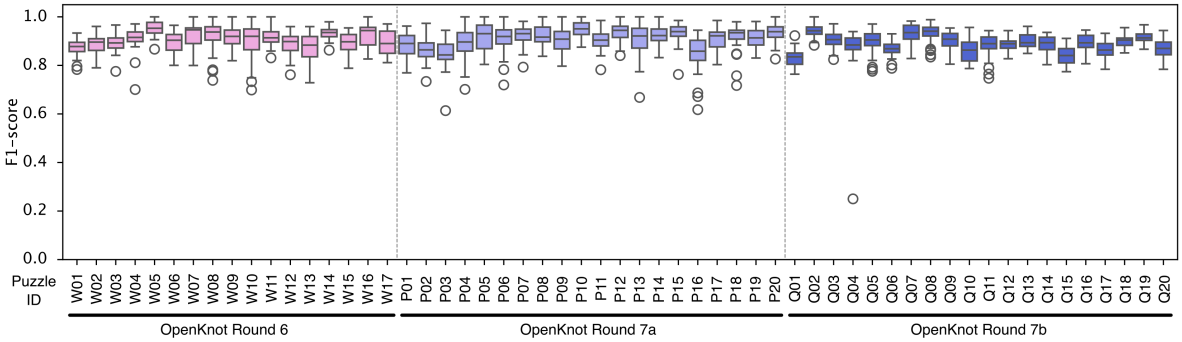

Figure S5 | Adherence to Base Pair Conditioning. Secondary-structure similarity between the target OpenKnot puzzle templates and RFDpoly diffusion outputs, quantified as the F1-score between *target* and *generated* base pair sets for each puzzle across OpenKnot rounds 6, 7a, and 7b. For each target structure, distributions summarize multiple (n=10) independent diffusion trajectories run with base pair template conditioning to guide the denoising process. To enable direct readout of secondary-structure satisfaction at the end of each trajectory, designs were generated using sequence–structure codesign, so that trajectories terminate with fully specified base identities and corresponding base coordinates.

E

What is a base pair?

As shown throughout this dissertation, base pair conditioning is a primary mechanism for controlling de novo nucleic-acid structure generation. Across RFdiffusion-polymer, RFdiffusion-3, and subsequent latent-diffusion extensions, this capability has depended on a reliable method for annotating base pairs within ground-truth structures during training.

When I began working on this problem, available base pair annotation tools were either not suitable for the noncanonical structural criteria relevant to our design tasks, closed-source, or both. As a result, I developed and iteratively refined a fully open mathematical framework for base pair identification.

This framework was re-implemented across three generations of the RFdiffusion stack and improved at each step: coarse-grained approximations became more precise, the pipeline became more general and robust to edge cases (including noncanonical nucleic acids and amino-acid interactions), and the geometric definitions were repeatedly stress-tested and tightened.

Given how quickly these models evolve, this system will likely be reimplemented again in future versions. This chapter is written as a practical reference for that future implementation: it provides a clear specification of the geometric quantities, their derivations, and the heuristic filters that have proven effective in our design setting.

E.1 Background and debate in the field

The term *base pair* predates the modern diversity of nucleic-acid structural biology, and its meaning has shifted depending on the scientific task at hand. In its original and most widely recognized sense, a "base pair" referred to complementary molecular interaction that stabilized the DNA double helix, as proposed in the Watson–Crick model [52]. In this context, base pairing was defined by specific complementarity (A–T and G–C) driven by hydrogen-bond patterns. Over time, however, it became clear that nucleobases can adopt alternative pairing geometries and still form stable, planar, hydrogen-bonded contacts that are functionally and structurally important.

As nucleic-acid structural biology matured, it became clear that restricting "base pairs" to canonical Watson–Crick complements was insufficient for many downstream goals: alternative hydrogen-bonded geometries (e.g. Hoogsteen)[53] and function-driven pairing rules in translation (wobble)[54] both demanded broader, task-dependent notions of what it means for two bases to "pair." This pressure was even stronger for structured RNA, where non-Watson–Crick interactions are not rare exceptions but recurrent, stereospecific contacts that help define tertiary motifs and global folds, motivating systematic classification schemes such as the Leontis–Westhof nomenclature [55]. In parallel, quantitative, coordinate-frame descriptions of bases enabled standard geometric parameters and analysis toolchains (e.g. NUPARM/NUCGEN, 3DNA, DSSR),[56, 57, 58] while annotation-oriented frameworks (e.g. FR3D) operationalized base pairing as a reproducible coordinate-labeling problem with explicit geometric and hydrogen-bond criteria [59].

Taken together, these developments explain why "base pair" remains a context-dependent term. In thermodynamics and secondary-structure prediction, the phrase often implicitly refers to Watson–Crick (and sometimes wobble) pairs because they dominate helix energetics and allow compact algorithmic abstractions [60]. In structural annotation and motif analysis, broader geometric and chemical definitions are required to capture non-canonical interactions that are essential to RNA tertiary architecture.[55, 58, 59] In this dissertation, we adopt an explicitly operational definition designed for *structure-guided generative modeling*: an interaction should be counted as a

base pair if it (i) is supported by a sufficiently strong hydrogen-bond network and (ii) exhibits near-planar base geometry with limited normal displacement, thereby serving as a reliable determinant of local rigidity and fold specification.

E.2 Why it matters for design

In this dissertation, base pair conditioning is used as a primary mechanism for imposing structural rigidity and stabilizing well-defined nucleic-acid folds. Motivated by the diversity of base–base interactions observed in native RNA and DNA, including many non-Watson–Crick–Franklin (WCF) contacts (**Supplementary Fig. S6**), I found it necessary to adopt an operational definition of a "base pair" that extends beyond canonical A–U/T and G–C patterns while remaining mechanically meaningful for design.

A key distinction between proteins and nucleic acids is where the dominant hydrogen-bonding that defines secondary structure occurs. In proteins, repetitive backbone hydrogen bonds between amide N–H and carbonyl C=O groups generate α-helices and β-sheets; once these backbone-mediated secondary-structure elements form, side chains can be used primarily to specify tertiary packing (hydrophobic cores, salt bridges, etc.). In nucleic acids, by contrast, the most structurally-defining hydrogen-bond networks are largely formed by the nucleobases themselves. Thus, the same chemical groups (the bases) are heavily "consumed" in establishing helical secondary structure, leaving fewer degrees of freedom available to independently encode tertiary interactions. As a result, tertiary rigidity often emerges from precisely placed helical interruptions (bulges, junctions, internal loops) and from additional stereospecific contacts such as strand/loop–groove clasping and helix–helix register docking.

From a design perspective, this makes base pairing the dominant source of structural definition and therefore a natural control knob for generative modeling and conditioning. However, to use base pairing as a general conditioning signal, we require an algorithmic definition that captures (i) the energetic signature of a base pair (multiple hydrogen bonds stabilizing a specific relative geometry)

and (ii) the geometric signature of a base pair (near-coplanar bases with limited displacement along the base-plane normal). The remainder of this chapter formalizes the operational definition used throughout this work when annotating structural datasets and constructing model input features.

E.3 Operational definition

For residues (tokens) i and j, we declare a base pair interaction using two coupled criteria:

1. **Hydrogen-bond network support:** residues i and j must form a sufficiently strong weighted count of hydrogen bonds involving base atoms (full weight) and coplanar backbone atoms (reduced weight).
2. **Planar geometry:** the bases must be approximately coplanar and exhibit limited displacement along the pair-frame normal; additionally, the relative tilts (buckle and propeller) must lie within empirically chosen bounds.

These criteria are deliberately *operational*: they are designed to robustly recover the set of interactions that behave as structurally defining base pairs in 3D structures (including many non-canonical cases), rather than to reproduce any single historical naming convention.

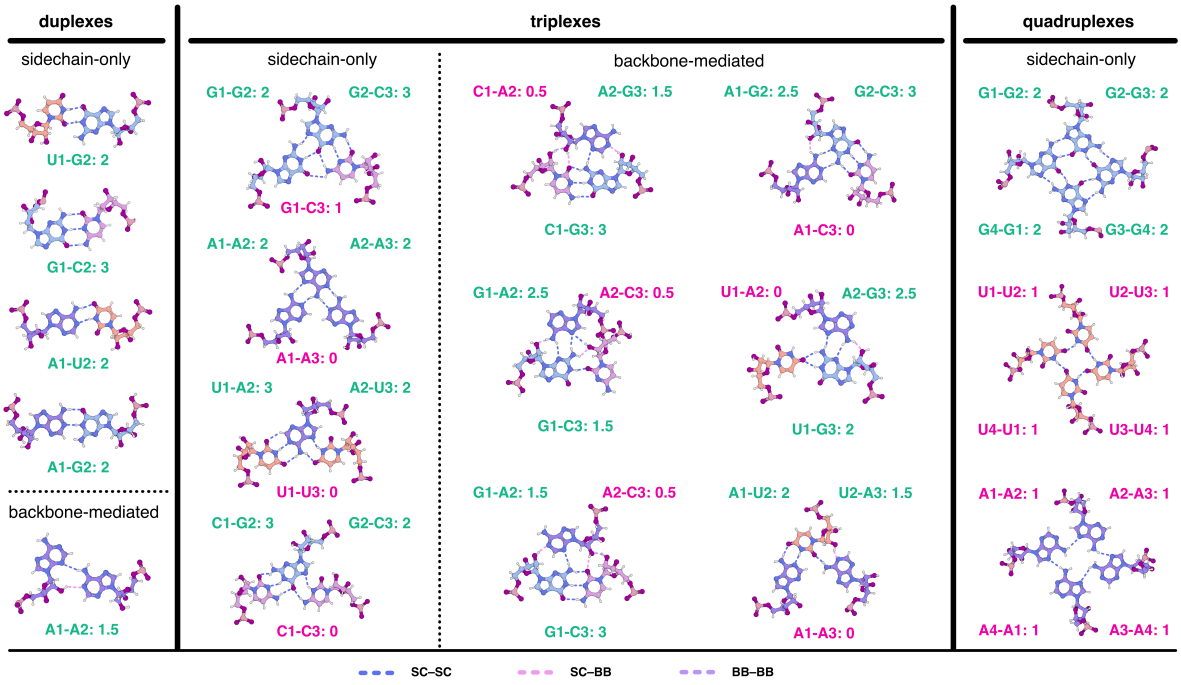

Figure S6 | Base pair examples and H-bond counting. Different types of base pairs, formed through different combinations of hydrogen bonds between sidechain and backbone functional groups. Summation of hydrogen bonds between two positions have their relative weights scaled according to the interaction types: sidechain-sidechain (1.0), sidechain-backbone (0.5), backbone-backbone (0.0) to produce a weighted count that is then transformed into a soft confidence score for base pair annotation and filtering. This count is then passed through a sigmoid function to produce a continuous confidence score for base pair annotation, scaled between 0 and 1.

Design goals. This annotation procedure was designed for diffusion conditioning and therefore prioritizes four practical goals: (i) *chemical interpretability* (each accepted pair is supported by explicit hydrogen-bond evidence and explicit geometric criteria); (ii) *robustness* to geometric imperfections or noise, so borderline but plausible interactions are not discarded prematurely; (iii) *portability* across structural representations as long as local frames and interaction criteria can be computed.

E.3.1 Hydrogen-bond network score

Let $H_{ij}^{(v)}$ denote the *count* of hydrogen bonds between residues i and j of interaction class v, where we separate three categories: $v \in \{\text{BB-BB},\ \text{BB-SC},\ \text{SC-SC}\}$, with "SC" indicating nucleobase (sidechain) atoms and "BB" indicating backbone atoms. We form a weighted summation

$$S_{ij} \;=\; \sum_{v} w_v\, H_{ij}^{(v)}, \qquad (w_{\text{BB-BB}} = 0,\, w_{\text{BB-SC}} = \frac{1}{2},\, w_{\text{SC-SC}} = 1)$$

The weighting reflects the design goal: base–base hydrogen bonds most directly stabilize a well-defined base pair geometry, base–backbone hydrogen bonds can contribute but are typically less "direct" (and may couple to additional backbone/ribose degrees of freedom), and backbone–backbone hydrogen bonds are not treated as base pair defining in this scheme.

We convert S_{ij} into a continuous satisfaction score via a logistic map:

$$p_{ij}^{\text{HB}} \;=\; \sigma\Big(\kappa\big(S_{ij} - (S_{\min} - 1)\big)\Big), \qquad \sigma(x) = \frac{1}{1 + e^{-x}}.$$

Here $S_{\min} = 2$ is the target "two base-like hydrogen bonds" scale, and κ is a steepness parameter (chosen heuristically). With this parameterization, $S_{ij} = 1$ yields $p_{ij}^{\text{HB}} \approx 0.5$, while $S_{ij} \gtrsim 2$ quickly saturates toward 1, matching the intuition that two base–base hydrogen bonds should strongly indicate a base pair.

When a hard decision is required, we binarize by selecting values meeting a chosen level of stringency. Typically $p_{ij}^{\text{HB}} > 0.9$ provides a binarization relevant to our design goals.

Note: Although not yet tested in prior implementations, I believe there is value in using non-binarized features and exposing fractional values of p_{ij}^{HB} as token-level pair features. Single hydrogen bonds may be insufficient to stabilize an isolated pair, but in cyclic support networks, cooperative effects can make such interactions structurally meaningful. This behavior appears in recent structures (e.g., PDB ID: 9WHV), which include U-quadruplex and A-quadruplex motifs where single

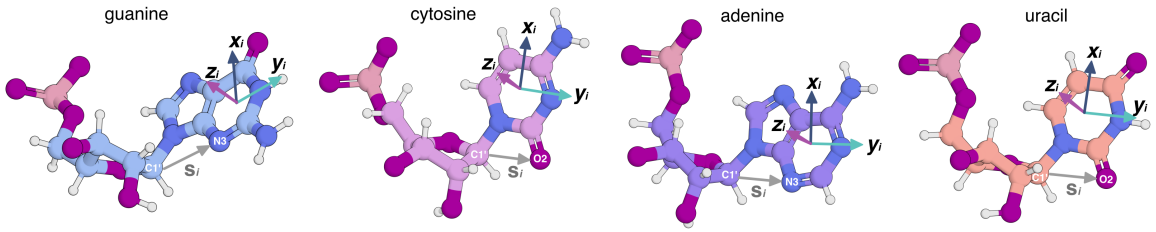

Figure S7 | The standard local frames of nucleotide sidechains Components denoted by vectors: $\mathbf{x}_i$, $\mathbf{y}_i$, and $\mathbf{z}_i$. Frames are produced using base–normal vectors $\mathbf{n}_i$, and sugar–edge vectors, $\mathbf{s}_i$.

hydrogen bonds per edge form stable four-partner rings (**Supplementary Fig. S6**). If we aim for finer control of these motifs, avoiding information loss from hard binarization is likely beneficial.

E.3.2 Geometric criteria of base pairing

Prior work has established principled geometric descriptions of base pairing and base-step parameters (e.g. 3DNA, NUPARM, DSSR).[10, 61, 62, 57, 56, 63]. To enable fully open-source annotation and to integrate geometry directly into our pipelines, we compute a set of geometric filters from atomic coordinates.

Local base frames (per residue). Let $\{\mathbf{r}_{i,k}\}_{k=1}^{K_i}$ be the coordinates of the planar nucleobase atoms for residue i (atoms given by $k \in K_i$), and let

$$\mathbf{c}_i \;=\; \frac{1}{K_i}\sum_{k=1}^{K_i}\mathbf{r}_{i,k}$$

be their centroid. We compute a covariance matrix

$$\mathbf{C}_i \;=\; \frac{1}{K_i-1}\sum_{k=1}^{K_i}(\mathbf{r}_{i,k}-\mathbf{c}_i)(\mathbf{r}_{i,k}-\mathbf{c}_i)^\top,$$

Since $\mathbf{C}_i$ is real symmetric, it admits an orthonormal eigendecomposition

$$\mathbf{C}_i = \mathbf{Q}_i\,\mathrm{diag}(\lambda_{i,1},\lambda_{i,2},\lambda_{i,3})\,\mathbf{Q}_i^\top, \qquad \lambda_{i,1}\le\lambda_{i,2}\le\lambda_{i,3},$$

where the columns $\mathbf{q}_{i,1}, \mathbf{q}_{i,2}, \mathbf{q}_{i,3}$ of $\mathbf{Q}_i$ are eigenvectors of $\mathbf{C}_i$. We define the (unnormalized) base-plane normal as

$$\mathbf{n}_i := \mathbf{q}_{i,1},$$

i.e., the eigenvector associated with the smallest eigenvalue of $\mathbf{C}_i$.

To make the base normals numerically consistent across the chain (as it turns out, bases pivot very frequently in order to form noncanonical interactions), we re-orient $\mathbf{n}_i$ based on the directionality of the backbone vector via projection. Let $\mathbf{r}_{i,\text{frame}}$ be a representative (frame-center) backbone coordinate ($C1'$ for nucleotides, $C\alpha$ for proteins), and define a smoothed backbone direction

$$\mathbf{M}_i \;=\; \tfrac{1}{2}\big((\mathbf{r}_{i,\text{frame}} - \mathbf{r}_{i-1,\text{frame}}) + (\mathbf{r}_{i+1,\text{frame}} - \mathbf{r}_{i,\text{frame}})\big).$$

Let $\widehat{\mathbf{n}}_i = \mathbf{n}_i/\|\mathbf{n}_i\|$ be the unit base-plane normal. We compute its signed scalar projection onto $\mathbf{M}_i$,

$$\alpha_i \;=\; \widehat{\mathbf{n}}_i^{\top}\mathbf{M}_i,$$

and define the oriented normal by scaling and renormalizing:

$$\mathbf{Z}_i \;=\; \frac{\alpha_i\,\widehat{\mathbf{n}}_i}{\|\alpha_i\,\widehat{\mathbf{n}}_i\|} \;=\; \frac{(\widehat{\mathbf{n}}_i^{\top}\mathbf{M}_i)\,\widehat{\mathbf{n}}_i}{\big|\widehat{\mathbf{n}}_i^{\top}\mathbf{M}_i\big|}.$$

Equivalently, $\mathbf{z}_i = \operatorname{sgn}(\widehat{\mathbf{n}}_i^{\top}\mathbf{M}_i)\,\widehat{\mathbf{n}}_i$, but the projection form highlights that we orient the normal to have positive component along the backbone direction.

To complete a right-handed local frame, we define a "sugar-edge" direction using two residue-type-dependent atoms: the "sugar-edge" vector, $\mathbf{s}_i$, begins at the $C1'$ atom for all nucleotides, but terminates at either the $N3$ atom for purines, or the $O2$ atom for pyrimidines (we will denote these the sugar–edge "start" and "stop" atoms, respectively):

$$\mathbf{s}_i \;=\; \frac{\mathbf{r}_{i,\text{stop}} - \mathbf{r}_{i,\text{start}}}{\|\mathbf{r}_{i,\text{stop}} - \mathbf{r}_{i,\text{start}}\|},$$

and with this direction defined, we can find orthonormal vectors between the sugar-edge and the base-normal using cross-products, and complete the remaining two components of our local base frames as follows:

$$\mathbf{x}_i \;=\; \frac{\mathbf{z}_i \times \mathbf{s}_i}{\|\mathbf{z}_i \times \mathbf{s}_i\|}, \qquad \mathbf{y}_i \;=\; \mathbf{x}_i \times \mathbf{z}_i.$$

yielding a complete set of local frame vectors to define the geometric placement of a nucleotide base (**Supplementary Fig. S7**).

Pair frame (per residue pair) Having defined per-position local frames, $x_i\, y_i\, z_i$, we next define pairwise frame representations, $X_{ij}\, Y_{ij}\, Z_{ij}$, which capture the semi-local geometry of position i relative to a potential partner, position j.

These pair frames allow geometric changes in single-position frames to be measured relative to a shared reference, providing a clean and numerically stable basis for defining pairwise geometric parameters.

The first key step is to define a shared pair normal by *constructively averaging* the two base normals. Our pair frame will primarily be defined by the shared mean base-normal for positions -i and -j, so we place most emphasis on numerical choices when performing this averaging. Because a plane normal is defined only up to sign (whether it points "above" or "below" a base plane depends on convention), two geometrically coplanar bases may have normals that are nearly parallel ($\mathbf{z}_i^\top \mathbf{z}_j \approx +1$) or nearly antiparallel ($\mathbf{z}_i^\top \mathbf{z}_j \approx -1$). We therefore flip $\mathbf{z}_j$ when necessary so that both normals point to the same hemisphere before averaging. Concretely, define

$$s_{ij} \;=\; \operatorname{sgn}\Big(\mathbf{z}_i^\top \mathbf{z}_j\Big) \in \{+1, -1\},$$

(with sgn(0) taken as +1), and set the pairwise normal to the normalized constructive mean:

$$\mathbf{Z}_{ij} \;=\; \frac{\mathbf{z}_i + s_{ij}\mathbf{z}_j}{\|\mathbf{z}_i + s_{ij}\mathbf{z}_j\|}.$$

Conceptually, $\mathbf{Z}_{ij}$ is obtained by choosing between $\mathbf{z}_i + \mathbf{z}_j$ (parallel case) and $\mathbf{z}_i - \mathbf{z}_j$ (antiparallel case) so as to maximize the magnitude of the average, ensuring a stable and consistent method for defining a plane-normal component for our shared pair frames.

For the next component of our pair frames, we consider a direction that points from the representative atoms of residue-i to residue-j to anchor the direction of the y–component of our pair frame.

We can define an initial approximation of the perpendicular y–component using the direction of displacement between two representative atoms.
Given displacement vector, $\mathbf{d}_{ij,\text{frame}} = \mathbf{r}_{j,\text{frame}} - \mathbf{r}_{i,\text{frame}}$, we produce the normalized direction approximation:

$$\tilde{\mathbf{Y}}_{ij} \;=\; \frac{\mathbf{d}_{ij,\text{frame}}}{\|\mathbf{d}_{ij,\text{frame}}\|} \;=\; \frac{\mathbf{r}_{j,\text{frame}} - \mathbf{r}_{i,\text{frame}}}{\|\mathbf{r}_{j,\text{frame}} - \mathbf{r}_{i,\text{frame}}\|},$$

From here, we can re-introduce $\mathbf{Z}_{ij}$ as our independent seed direction, and produce the final two orthonormal components of our pair frame:

$$\mathbf{X}_{ij} \;=\; \frac{\mathbf{Z}_{ij} \times \tilde{\mathbf{Y}}_{ij}}{\|\mathbf{Z}_{ij} \times \tilde{\mathbf{Y}}_{ij}\|}, \qquad \mathbf{Y}_{ij} \;=\; \frac{\mathbf{X}_{ij} \times \mathbf{Z}_{ij}}{\|\mathbf{X}_{ij} \times \mathbf{Z}_{ij}\|},$$

Note: I considered two choices for the seed axis used to construct orthogonal pair-frame bases: Y_{ij} (inter-base displacement) and Z_{ij} (mean base normal). I chose Z_{ij} as the primary reference because the angular parameters (*buckle* and *propeller*) are most naturally interpreted as deviations of local base normals from a shared normal axis.

An alternative interpretation, where propeller is treated as rotation about the inter-base dis-

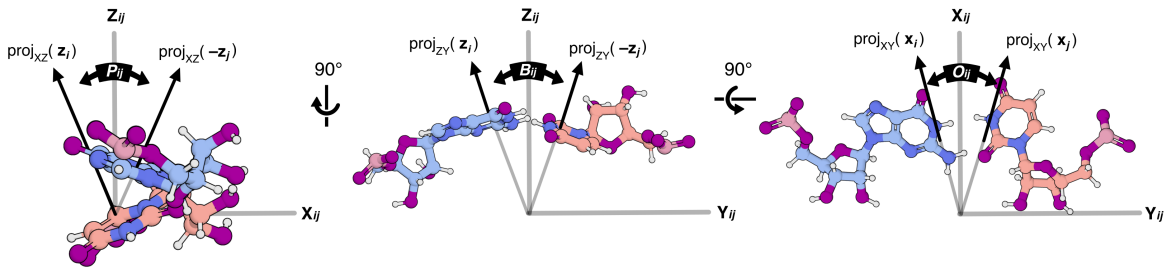

Figure S8 | Angular geometry parameters within pair frames These parameters describe the principal modes of relative orientation between two nucleotide bases. Propeller, P_{ij}, is defined within the span of the $Z--X$ plane; buckle, B_{ij}, is defined within the span of the $Z--Y$ plane; opening, O_{ij}, is defined within the span of the $X--Y$ plane.

placement axis, is also valid. However, the formulation used here provides better sensitivity to deviations around the shared base-normal reference and proved more numerically stable in all-by-all pair geometry computations.

Related formulations appear in prior work [56, 63]. In practice, those formulations may be best suited for evaluating already-identified local base pairs, whereas the equations used here are tuned for robust filtering over broader pair sets, including distant and highly misoriented candidates.

Angular parameters (buckle, propeller, opening). Each angular parameter is defined by the same recipe: choose one of the three pair-frame coordinate planes (YZ, XZ, or XY), project the relevant local-frame axes into that plane, and compute the angle between the resulting projected directions. In this sense, the definitions are cyclic over pair-frame planes: buckle uses the YZ plane, propeller uses the XZ plane, and opening uses the XY plane (**Supplementary Fig. S8**).

1. **Buckle angle.** Buckle measures how the two base normals differ *within the pair-frame YZ plane*, i.e. within $\mathrm{span}\{\mathbf{Y}_{ij}, \mathbf{Z}_{ij}\}$. Let proj_{YZ} denote orthogonal projection onto $\mathrm{span}\{\mathbf{Y}_{ij}, \mathbf{Z}_{ij}\}$:

$$\mathrm{proj}_{YZ}(\mathbf{v}) = (\mathbf{v}^\top \mathbf{Y}_{ij})\mathbf{Y}_{ij} + (\mathbf{v}^\top \mathbf{Z}_{ij})\mathbf{Z}_{ij}.$$

Define

$$\widehat{\mathbf{z}}^{(B)}_{i\to ij} = \frac{\mathrm{proj}_{YZ}(\mathbf{z}_i)}{\|\mathrm{proj}_{YZ}(\mathbf{z}_i)\|}, \qquad \widehat{\mathbf{z}}^{(B)}_{j\to ij} = \frac{\mathrm{proj}_{YZ}(\mathbf{z}_j)}{\|\mathrm{proj}_{YZ}(\mathbf{z}_j)\|},$$

and compute

$$B_{ij} = \arccos\Big(\widehat{\mathbf{z}}^{(B)}_{i\to ij} \cdot \big(-\widehat{\mathbf{z}}^{(B)}_{j\to ij}\big)\Big).$$

2. **Propeller angle.** Propeller measures how the two base normals differ *within the pair-frame XZ plane*, i.e. within $\mathrm{span}\{\mathbf{X}_{ij}, \mathbf{Z}_{ij}\}$. Let proj_{XZ} denote orthogonal projection onto $\mathrm{span}\{\mathbf{X}_{ij}, \mathbf{Z}_{ij}\}$:

$$\mathrm{proj}_{XZ}(\mathbf{v}) = (\mathbf{v}^\top \mathbf{X}_{ij})\mathbf{X}_{ij} + (\mathbf{v}^\top \mathbf{Z}_{ij})\mathbf{Z}_{ij}.$$

Define

$$\widehat{\mathbf{z}}^{(P)}_{i\to ij} = \frac{\mathrm{proj}_{XZ}(\mathbf{z}_i)}{\|\mathrm{proj}_{XZ}(\mathbf{z}_i)\|}, \qquad \widehat{\mathbf{z}}^{(P)}_{j\to ij} = \frac{\mathrm{proj}_{XZ}(\mathbf{z}_j)}{\|\mathrm{proj}_{XZ}(\mathbf{z}_j)\|},$$

and compute

$$P_{ij} = \arccos\Big(\widehat{\mathbf{z}}^{(P)}_{i\to ij} \cdot \big(-\widehat{\mathbf{z}}^{(P)}_{j\to ij}\big)\Big).$$

3. **Opening angle.** Opening measures the relative rotation of the in-plane base axes *within the pair-frame XY plane*, i.e. within $\mathrm{span}\{\mathbf{X}_{ij}, \mathbf{Y}_{ij}\}$. Let proj_{XY} denote orthogonal projection onto $\mathrm{span}\{\mathbf{X}_{ij}, \mathbf{Y}_{ij}\}$:

$$\mathrm{proj}_{XY}(\mathbf{v}) = (\mathbf{v}^\top \mathbf{X}_{ij})\mathbf{X}_{ij} + (\mathbf{v}^\top \mathbf{Y}_{ij})\mathbf{Y}_{ij}.$$

Define

$$\widehat{\mathbf{x}}^{(O)}_{i\to ij} = \frac{\mathrm{proj}_{XY}(\mathbf{x}_i)}{\|\mathrm{proj}_{XY}(\mathbf{x}_i)\|}, \qquad \widehat{\mathbf{x}}^{(O)}_{j\to ij} = \frac{\mathrm{proj}_{XY}(\mathbf{x}_j)}{\|\mathrm{proj}_{XY}(\mathbf{x}_j)\|},$$

and compute

$$O_{ij} = \arccos\Big(\widehat{\mathbf{x}}^{(O)}_{i\to ij} \cdot \widehat{\mathbf{x}}^{(O)}_{j\to ij}\Big).$$

Note: While we show the calculation of the "opening" angle here, in practice we use it solely for diagnostic classification, but not as a filter for base pair classification. Opening varies substantially across interaction chemistries (Watson–Crick, Hoogsteen, sugar-edge, and mixed-edge contacts), so enforcing a narrow opening range would systematically exclude legitimate non-canonical interactions that are structurally useful in tertiary motifs.

Vertical displacement (rise). In addition to angular parameters, we use a scalar coplanarity proxy that measures how far the two bases are displaced along the shared pair normal. Let $\mathbf{d}_{ij}^{\mathrm{sc}} = \mathbf{c}_j - \mathbf{c}_i$ be the displacement between base centroids. The signed normal displacement is

$$H_{ij} \;=\; (\mathbf{d}_{ij}^{\mathrm{sc}})^{\top}\mathbf{Z}_{ij},$$

where $|H_{ij}|$ is small when the base centroids lie near a common plane orthogonal to $\mathbf{Z}_{ij}$.

Calibration and transfer. The weights and thresholds in Eqs. (E.3.1)–(E.3.3) should be interpreted as calibrated heuristics for the present representation and hydrogen-bond detector, not universal physical constants. When porting the procedure to alternative coordinate sets (e.g. coarse-grained sites, predicted structures, modified donor/acceptor definitions), the recommended practice is to preserve the algorithmic structure while re-tuning these parameters on a curated validation set.

Sequence-neighborhood exclusion. To avoid labeling trivial local contacts as base pairs, we exclude residue pairs within a small chain-local neighborhood (chain-aware), i.e. we require $|i-j| > \delta_{\mathrm{seq}}$ (or the analogous chain-indexed distance) before considering (i, j) eligible for pairing.

E.3.3 Final base pair decision

From our pairwise geometry parameters, we have a set of acceptable ranges that positions (i, j) must fall within in order to be geometrically consistent with base pairing:

$$|H_{ij}| \leq 1.5\mathring{A}, \qquad B_{ij} \in [0, \frac{\pi}{5}] \cup [\frac{4\pi}{5}, \pi], \qquad P_{ij} \in [0, \frac{\pi}{5}] \cup [\frac{4\pi}{5}, \pi],$$

Combined with the hydrogen-bond threshold condition, $p_{ij}^{\mathrm{HB}} > 0.9$, these ranges define a boolean mask used to assign base pair presence.

Geometry vs. conditioning policy. Operationally, this procedure separates *geometric validity* from *conditioning policy*. The hydrogen-bond and geometry filters define which residue pairs are physically plausible, while the downstream `MASK`/`PAIR`/`LOOP` encoding determines what information is exposed to the model during training or inference. This decoupling allows supervision sparsity and masking strategy to be varied without redefining the underlying structural annotation algorithm.

Bibliography

[1] Sefah, K., Shangguan, D., Xiong, X., O'donoghue, M. B. & Tan, W. Development of dna aptamers using cell-selex. *Nature protocols* **5**, 1169–1185 (2010).

[2] Kang, K.-N. & Lee, Y.-S. Rna aptamers: a review of recent trends and applications. *Future Trends in Biotechnology* 153–169 (2012).

[3] Lee, J. *et al.* Rna design rules from a massive open laboratory. *Proceedings of the National Academy of Sciences* **111**, 2122–2127 (2014).

[4] Zadeh, J. N. *et al.* Nupack: Analysis and design of nucleic acid systems. *Journal of computational chemistry* **32**, 170–173 (2011).

[5] Poppleton, E. *et al.* Rna origami: design, simulation and application. *RNA biology* **20**, 510–524 (2023).

[6] Zadegan, R. M. & Norton, M. L. Structural dna nanotechnology: from design to applications. *International journal of molecular sciences* **13**, 7149–7162 (2012).

[7] Rothemund, P. W. Folding dna to create nanoscale shapes and patterns. *Nature* **440**, 297–302 (2006).

[8] Seeman, N. C. Nucleic acid junctions and lattices. *Journal of theoretical biology* **99**, 237–247 (1982).

[9] Geary, C., Grossi, G., McRae, E. K., Rothemund, P. W. & Andersen, E. S. Rna origami design tools enable cotranscriptional folding of kilobase-sized nanoscaffolds. *Nature chemistry* **13**, 549–558 (2021).

[10] Lu, X.-J. & Olson, W. K. 3dna: a software package for the analysis, rebuilding and visualization of three-dimensional nucleic acid structures. *Nucleic acids research* **31**, 5108–5121 (2003).

[11] Li, S., Olson, W. K. & Lu, X.-J. Web 3dna 2.0 for the analysis, visualization, and modeling of 3d nucleic acid structures. *Nucleic acids research* **47**, W26–W34 (2019).

[12] Yesselman, J. D. *et al.* Computational design of three-dimensional rna structure and function. *Nature nanotechnology* **14**, 866–873 (2019).

[13] Baek, M. *et al.* Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871–876 (2021).

[14] Wang, J. *et al.* Scaffolding protein functional sites using deep learning. *Science* **377**, 387–394 (2022).

[15] Watson, J. L. *et al.* De novo design of protein structure and function with rfdiffusion. *Nature* **620**, 1089–1100 (2023).

[16] Krishna, R. *et al.* Generalized biomolecular modeling and design with rosettafold all-atom. *Science* **384**, eadl2528 (2024).

[17] Baek, M. *et al.* Accurate prediction of protein–nucleic acid complexes using rosettafoldna. *Nature methods* **21**, 117–121 (2024).

[18] Ma, R. *et al.* Riboflow: Conditional de novo rna sequence-structure co-design via synergistic flow matching. *arXiv preprint arXiv:2503.17007* (2025).

[19] Nori, D. & Jin, W. Rnaflow: Rna structure & sequence design via inverse folding-based flow matching. *arXiv preprint arXiv:2405.18768* (2024).

[20] Morehead, A., Ruffolo, J., Bhatnagar, A. & Madani, A. Towards joint sequence-structure generation of nucleic acid and protein complexes with se (3)-discrete diffusion. *arXiv preprint arXiv:2401.06151* (2023).

[21] Anand, R. *et al.* Rna-frameflow: Flow matching for de novo 3d rna backbone design. *ArXiv* arXiv–2406 (2025).

[22] Tarafder, S. & Bhattacharya, D. Rnabpflow: Base pair-augmented se (3)-flow matching for conditional rna 3d structure generation. *bioRxiv* (2025).

[23] Kubaney, A. *et al.* Rna sequence design and protein–dna specificity prediction with na-mpnn. *bioRxiv* 2025–10 (2025).

[24] Dauparas, J. *et al.* Robust deep learning–based protein sequence design using proteinmpnn. *Science* **378**, 49–56 (2022).

[25] Dauparas, J. *et al.* Atomic context-conditioned protein sequence design using ligandmpnn. *Nature Methods* 1–7 (2025).

[26] Cruz, J. A. *et al.* Rna-puzzles: a casp-like evaluation of rna three-dimensional structure prediction. *Rna* **18**, 610–625 (2012).

[27] Eterna openknot challenge. https://eternagame.org/challenges/11843006 (2025).

[28] He, S. *et al.* Ribonanza: deep learning of rna structure through dual crowdsourcing. *bioRxiv* (2024).

[29] Kretsch, R. C. *et al.* Assessment of nucleic acid structure prediction in casp16. *Proteins: Structure, Function, and Bioinformatics* (2025).

[30] Boitreaud, J. *et al.* Chai-1: Decoding the molecular interactions of life. *BioRxiv* (2024).

[31] Abramson, J. *et al.* Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature* **630**, 493–500 (2024).

[32] Das, R. *et al.* Assessment of three-dimensional rna structure prediction in casp15. *Proteins: Structure, Function, and Bioinformatics* **91**, 1747–1770 (2023).

[33] Zemla, A. Lga: a method for finding 3d similarities in protein structures. *Nucleic acids research* **31**, 3370–3374 (2003).

[34] Zhang, C., Shine, M., Pyle, A. M. & Zhang, Y. Us-align: universal structure alignments of proteins, nucleic acids, and macromolecular complexes. *Nature methods* **19**, 1109–1115 (2022).

[35] Marinus, T., Fessler, A. B., Ogle, C. A. & Incarnato, D. A novel shape reagent enables the analysis of rna structure in living cells with unprecedented accuracy. *Nucleic acids research* **49**, e34–e34 (2021).

[36] Wayment-Steele, H. K. *et al.* Rna secondary structure packages evaluated and improved by high-throughput experiments. *Nature methods* **19**, 1234–1242 (2022).

[37] Mortimer, S. A., Trapnell, C., Aviran, S., Pachter, L. & Lucks, J. B. Shape–seq: high-throughput rna structure analysis. *Current protocols in chemical biology* **4**, 275–297 (2012).

[38] Das, R. Openknotscorematlab. https://github.com/eternagame/OpenKnotScoreMATLAB (2025).

[39] Croll, T. I. Isolde: a physically realistic environment for model building into low-resolution electron-density maps. *Biological Crystallography* **74**, 519–530 (2018).

[40] Glasscock, C. J. *et al.* Computational design of sequence-specific dna-binding proteins. *Nature Structural & Molecular Biology* **32**, 2252–2261 (2025).

[41] Holliday, R. A mechanism for gene conversion in fungi. *Genetics Research* **5**, 282–304 (1964).

[42] Praetorius, F. & Dietz, H. Self-assembly of genetically encoded dna-protein hybrid nanoscale shapes. *Science* **355**, eaam5488 (2017).

[43] Joung, J. K. & Sander, J. D. Talens: a widely applicable technology for targeted genome editing. *Nature reviews Molecular cell biology* **14**, 49–55 (2013).

[44] Bhakta, M. S. & Segal, D. J. in *The generation of zinc finger proteins by modular assembly* (eds Mackay, J. P. & Segal, D. J.) *Engineered Zinc Finger Proteins: Methods and Protocols* Methods in Molecular Biology, 3–30 (Springer, Totowa, NJ, 2010).

[45] Kwon, D. Rna function follows form-why is it so hard to predict? *Nature* **639**, 1106–1108 (2025).

[46] Lauko, A. *et al.* Computational design of serine hydrolases. *Science* **388**, eadu2454 (2025).

[47] Yang, J. *et al.* Improved protein structure prediction using predicted interresidue orientations. *Proceedings of the National Academy of Sciences* **117**, 1496–1503 (2020).

[48] Tran, L. *et al.* Design of orthogonal far-red, orange and green fluorophore-binding proteins for multiplex imaging. *bioRxiv* (2025).

[49] Kretsch, R. C. *et al.* Naturally ornate rna-only complexes revealed by cryo-em. *Nature* 1–3 (2025).

[50] Punjani, A., Rubinstein, J. L., Fleet, D. J. & Brubaker, M. A. cryosparc: algorithms for rapid unsupervised cryo-em structure determination. *Nature methods* **14**, 290–296 (2017).

[51] Adamczyk, B., Antczak, M. & Szachniuk, M. Rnasolo: a repository of cleaned pdb-derived rna 3d structures. *Bioinformatics* **38**, 3668–3670 (2022).

[52] Watson, J. D., Crick, F. *et al.* A structure for deoxyribose nucleic acid (1953).

[53] Hoogsteen, K. The structure of crystals containing a hydrogen-bonded complex of 1-methylthymine and 9-methyladenine. *Acta crystallographica* **12**, 822–823 (1959).

[54] Crick, F. H. *et al.* Codon-anticodon pairing: the wobble hypothesis. *J. mol. Biol* **19**, 548–555 (1966).

[55] Leontis, N. B. & Westhof, E. Geometric nomenclature and classification of rna base pairs. *Rna* **7**, 499–512 (2001).

[56] Bansal, M., Bhattacharyya, D. & Ravi, B. Nuparm and nucgen: software for analysis and generation of sequence dependent nucleic acid structures. *Bioinformatics* **11**, 281–287 (1995).

[57] Lu, X.-J. & Olson, W. K. 3dna: a versatile, integrated software system for the analysis, rebuilding and visualization of three-dimensional nucleic-acid structures. *Nature protocols* **3**, 1213–1227 (2008).

[58] Lu, X.-J., Bussemaker, H. J. & Olson, W. K. Dssr: an integrated software tool for dissecting the spatial structure of rna. *Nucleic acids research* **43**, e142–e142 (2015).

[59] Sarver, M., Zirbel, C. L., Stombaugh, J., Mokdad, A. & Leontis, N. B. Fr3d: finding local and composite recurrent structural motifs in rna 3d structures. *Journal of mathematical biology* **56**, 215–252 (2008).

[60] Delisi, C. & Crothers, D. M. Prediction of rna secondary structure. *Proceedings of the National Academy of Sciences* **68**, 2682–2685 (1971).

[61] Das, J., Mukherjee, S., Mitra, A. & Bhattacharyya, D. Non-canonical base pairs and higher order structures in nucleic acids: crystal structure database analysis. *Journal of Biomolecular Structure and Dynamics* **24**, 149–161 (2006).

[62] Mukherjee, S., Bansal, M. & Bhattacharyya, D. Conformational specificity of non-canonical base pairs and higher order structures in nucleic acids: crystal structure database analysis. *Journal of computer-aided molecular design* **20**, 629–645 (2006).

[63] Bhattacharyya, D. & Bansal, M. A self-consistent formulation for analysis and generation of non-uniform dna structures. *Journal of Biomolecular Structure and Dynamics* **6**, 635–653 (1989).